CHARUSAT

CHAROTAR UNIVERSITY OF SCIENCE AND TECHNOLOGY

Faculty of Technology & Engineering

Bachelor of Technology Programme

(B.Tech. CE/CSE/IT)

ACADEMIC YEAR 2025-26

ACADEMIC REGULATIONS & SYLLABUS

(Choice Based Credit System)

Faculty of Technology and Engineering

B. Tech. Programme

Chandubhai S. Patel Institute of Technology

Devang Patel Institute of Advance Technology and Research

CHARUSAT
CHAROTAR UNIVERSITY OF SCIENCE AND TECHNOLOGY

FACULTY OF TECHNOLOGY AND ENGINEERING

ACADEMIC REGULATIONS

Bachelor of Technology Programmes

Choice Based Credit System

Academic regulations recommendations are provided to ensure uniform system of education, programmes duration, eligibility criteria for admission, course credits distribution, teaching and examination pedagogy, detailed syllabus with reference material.

1) System of Education

The Charotar University of Science and Technology (CHARUSAT) shall follow Choice based Credit System (CBCS) with Semester pattern at Undergraduate and Master levels. Each semester will be at least of 90 working days. Apart from the programme core courses, provision for choosing University level electives and Programme electives are available under the CBCS.

2) Duration of Programme

i) Undergraduate Programme (B.Tech.)

Minimum	8 semesters (4 academic years)
Maximum	16 semesters (8 academic years)

3) Eligibility for admissions

As enacted by Government of Gujarat/AICTE/UGC from time to time.

4) Mode of admissions

As enacted by Government of Gujarat from time to time.

5) Programme Structure and Credits

As per annexure – 1 attached

6) Attendance

6.1. Students are expected to maintain 100% attendance in all courses. However, students may involuntarily have to miss classes due to illness or some family emergency; students are permitted to maintain a minimum attendance of 75% with producing proof or

reason for the absence. In case of medical exigencies, the student/parent should inform the principal immediately through call or by email. Within a week, starting from the day of absence, the proof of medical exigency must be submitted to the Principal's office.

6.2. Unauthorized absence will be considered as part of the discretionary 25% for fulfilling the minimum 75% attendance requirement for appearing in the examination.

6.3. Students nominated/sponsored by the University to represent in various forums like seminars/conferences/workshops/competitions or taking part in co-curricular/extra-curricular events will be given attendance credit provided the student applies in writing for such a leave in advance and obtains sanction from the Principal of his/her Institute for academic related requests.

7) Course Evaluation

7.1. The performance of every student in each course will be evaluated as follows:

7.1.1 Internal evaluation by the course faculty member(s) based on continuous assessment. The respective department /institute will conduct the continuous assessment. The course faculty members shall share the pedagogy related to the continuous evaluation with the students.

7.1.2 Final end-semester examination shall be conducted by the University through written paper, practical test, oral test, presentation by the student or a combination of these.

7.1.3 The weightages of continuous assessment and end-semester university examination in overall assessment shall depend on individual course as approved by Academic Council through Faculty Board and Board of Studies.

7.1.4 The performance of candidate in continuous assessment and in end-semester examination together (if applicable) shall be considered for deciding the final grade in a course.

7.1.5 In order to earn the credit in a course a student has to obtain grade other than FF.

7.2. Performance in continuous assessment and end-semester University Examination

7.2.1 Minimum performance with respect to continuous assessment as well as end-semester university examination will be an important consideration for passing a course.

7.2.2 If a candidate fails to obtain minimum required overall percentage of marks (36%), student has to repeat the examination till the minimum required overall percentage obtained.

8) Grade Point System

8.1. The total of the internal evaluation marks and end semester examination marks in each course will be converted to a letter grade on a ten-point scale as per the following scheme:

8.2. Proposed Grading Scheme to awarding letter grade and grade point as per NEP 2020

Letter Grade	Grade Point	Grading Scheme for Mark (In %)
O (Outstanding)	10	96.0-100
A+ (Excellent)	9	86.0-95.9
A (Very Good)	8	76.0-85.9
B+ (Good)	7	66.0-75.9
B (Above Average)	6	56.0- 65.9
C (Average)	5	46.0 – 55.9
P (Pass)	4	36.0 – 45.9
F (Fail)	0	Below 36.0
Ab (Absent)	0	Absent

- The minimum passing marks for each pattern of evaluation are 36%

8.3. The student's performance in any semester will be assessed by the Semester Grade Point Average (SGPA). Similarly, his/her performance at the end of two or more consecutive semesters will be denoted by the Cumulative Grade Point Average (CGPA). The SGPA and CGPA are calculated as follows:

(i) SGPA = $\sum Ci\ Gi / \sum Ci$ where Ci is the number of credits of course i

Gi is the Grade Point for the course i

and i = 1 to n, n = number of courses in the semester

(ii) CGPA = $\sum$ Ci Gi / $\sum$ Ci where Ci is the number of credits of course i

Gi is the Grade Point for the course i

and i = 1 to n, n = number of courses of all semesters up to which CGPA is computed.

9) Award of Class

- The class awarded to a student in the programme is decided by the final CGPA as per the following scheme:

Award of Class	CGPA Range
First Class with Distinction	CGPA ≥ 7.0 & ≤ 10.0
First class	CGPA ≥ 6.0 & < 7.0
Second Class	CGPA ≥ 5.0 & < 6.0
Pass Class	CGPA < 5.0

10)Detention Criteria

- A student will be promoted to next year only if he/she has cleared all the courses of the year he/she is studying in.

  Link: https://charusat.ac.in/ => Student's Corner => Detention Rules

CHAROTAR UNIVERSITY OF SCIENCE AND TECHNOLOGY

CHAROTAR UNIVERSITY OF SCIENCE AND TECHNOLOGY (CHARUSAT)

FACULTY OF TECHNOLOGY & ENGINEERING (FTE)

CHOICE BASED CREDIT SYSTEM

FOR

BACHELOR OF TECHNOLOGY & ENGINEERING

(B.Tech. CE/CSE/IT)

Choice Based Credit System

With the aim of incorporating the various guidelines initiated by the University Grants Commission (UGC) to bring equality, efficiency and excellence in the Higher Education System, Choice Based Credit System (CBCS) has been adopted. CBCS offers wide range of choices to students in all semesters to choose the courses based on their aptitude and career objectives. It accelerates the teaching-learning process and provides flexibility to students to opt for the courses of their choice and / or undergo additional courses to strengthen their Knowledge, Skills and Attitude.

1. CBCS – Conceptual Definitions / Key Terms (Terminologies)

Types of Courses: The Programme Structure consist Foundation courses, Core courses, Elective courses, Non-credit (audit) courses and SWAYAM MOOCs.

1.1 Foundation Course

These courses are offered by the institute in order to prepare students for studying courses to be offered at higher levels.

1.2 Core Courses

A Course which shall compulsorily be studied by a candidate to complete the requirements of a degree / diploma in a said programme of study is defined as a core course. Following core courses are incorporated in CBCS structure:

A. University Core courses(UC):

University core courses are compulsory courses which are offered across university and must be completed in order to meet the requirements of programme. Environmental science will be a compulsory University core for all Undergraduate Programmes.

B. Programme Core courses(PC):

Programme core courses are compulsory courses offered by respective programme owners, which must be completed in order to meet the requirements of programme.

1.3 Elective Courses

Generally, a course which can be chosen from a pool of courses and which may be very specific or specialised or advanced or supportive to the discipline of study or which provides an extended scope or which enables an exposure to some other discipline / domain or nurtures the candidates proficiency / skill is called an elective course. Following elective courses are incorporated in CBCS structure:

A. University Elective Courses(UE):

The pool of elective courses offered across all faculties / programmes. As a general guideline, Programme should incorporate 2 University Electives of 2 credits each (total 4 credits).

B. Programme Elective Courses(PE):

The programme specific pool of elective courses offered by respective programme.

1.4 Non Credit Course (NC) - AUDIT Course

A 'Non Credit Course' is a course where students will receive Participation or Course Completion certificate. This will be reflected in Student's Grade Sheet but the grade of the course will not be considered to calculate SGPA and CGPA. Attendance and Course Assessment is compulsory for Non Credit Courses.

1.5 Credit Transfer through SWAYAM MOOCs

CHARUSAT provides credit transfer as per UGC guidelines to all the students from SWAYAM against elective courses. The credit transfer is offered in two modes: (a) Partial credit transfer (b) Full credit transfer.

1.6 Medium of Instruction

The Medium of Instruction will be English.

In consonance with the National Education Policy (NEP) 2020 and the guidelines of the University Grants Commission (UGC), Charotar University of Science and Technology (CHARUSAT) implements the Multiple Exit scheme in their Bachelor of Technology programme.

Facilitating multiple exit options with UG certificate/UG diploma/or UG degree depending upon the number of credits secured. Skill component with progressive enhancement in skills in respective disciplines is introduced in the curriculum right from the 1st year of the program to ensure the student employability after every exit.

Students may be permitted to take a break from the study during the period of study but the total duration

for completing the programme shall not exceed 7 years. If student wishes, he/she shall be allowed to re-enter the degree programme within three years and complete the degree programme within the stipulated maximum period of seven years. Table 1 shows the exit qualifications along with credit requirements.

Table 1 Exit Qualifications along with Credit Requirements

ACADEMIC LEVEL	EXIT QUALIFICATION AND CREDITS REQUIRED	NATIONAL CREDIT LEVEL (NCrF)
1st year of UG Degree	UG Certificate will be awarded Minimum 40 credits followed by an exit 4-credit skills enhancement course	4.5
2nd year of UG Degree	UG Diploma will be awarded Minimum of 80 credits followed by an exit 4-credit skills enhancement course	5.0
3rd year of UG Degree	B.Sc. will be awarded Minimum of 120 credits	5.5
4th year of UG Degree	B. Tech. will be awarded Minimum of 160 credits, with minimum of 40 credits each at level	6.0

Guidelines for multiple exit along with awarding UG Certificate, UG Diploma, and Degrees:

A. Exit after First Year

Award: UG Certificate in XXXX

Student must undertake a 4-Credits skill enhancement course before commencement of 2nd year. This shall require 120 hours engagement in the relevant industry/organization. Student shall have an option to continue the 2nd year onward study.

The student shall be awarded with "UG Certificate in XXXX", with redemption of credits from ABC. Total credits redemption shall be 1st year credits + 4 credits earned in summer.

B. Exit after Second Year

Award: UG Diploma in XXXX

Student must undertake a 4-Credits summer internship/ skill enhancement course before commencement of 3rd year. This shall require 120 hours engagement in the relevant industry/organization. Student shall have an option to continue the 3rd year onward study.

The student shall be awarded with "UG Diploma in XXXX", with redemption of credits from ABC. Total credits redemption shall be 1st and 2nd year credits + 4 credits earned in summer.

C. Exit after Third Year

Award: B.Sc. Degree in XXXX

The student shall be awarded with "B.Sc. Degree in XXXX", with redemption of 3 years' credits from ABC.

4-Credit (120 hrs.) Skills Enhancement Courses:

Level/ Branch	CE	IT	CSE	AIML
1st year of UG Degree	CEUS101: Web Development	ITUS102: Introduction of Web Technologies	CSUS101: Data Analysis	AIUS101: Data Analysis

Value Added Courses:

Inclusion of 2 credits courses on Community Service/ NSS/NCC/ Sports; and provision to earn extra credits based on undertaking Research/ Academic/Cultural/ and other Developmental activities is introduced.

- The component of 'Value-added Courses' could be enriched to include alternatives that could contribute to the holistic development of the students. In light of this, it is proposed to include a 02 credit Course on Community Service/ NSS/NCC/Sports as a compulsory course in the existing curricula offered across all UG programs at CHARUSAT. It is mandatory for all UG students (2024-25 admission batch) to undertake this course and the course shall not account for the overall CGPA.

- Provision to earn extra credits (**Skill Augmentation** course) based on co-curricular & extra-curricular activities is proposed. It was emphasized that the extra credits could help in fuller realization of the Graduate Attributes laid down by the university as well as could serve as an important ingredient facilitating the progression and recognition of fast learners. It is mandatory for all UG students (2024-25 admission batch) to undertake this course during the study and the credits of the course shall not account for the overall CGPA.

CHAROTAR UNIVERSITY OF SCIENCE & TECHNOLOGY (CHARUSAT)

Faculty of Technology & Engineering

Teaching & Examination Scheme for B. Tech Programme in CE/CSE/IT Year 2025-26

Sem	Course Code	Course Title	Teaching Scheme					Examination Scheme				
			Contact Hours		Credit		Credit	Theory		Practical		Total
			Theory	Practical	Theory	Practical		Internal	External	Internal	External	
SY Sem-3	ITUC201	Fundamentals of Computer Networks	4	2	4	1	5	50	50	25	25	150
	CSUC201	Fundamentals of Data Structure and Algorithms	3	2	3	1	4	50	50	25	25	150
	MSUD203	Discrete Mathematics	3	0	3	0	3	50	50	0	0	100
	ITUE203	Web Development Frameworks	3	4	3	2	5	50	50	25	25	150
	XXXXX	University Elective - I	0	4	0	2	2	0	0	25	25	50
	CEUE203	Object Oriented Programming	2	2	2	1	3	25	25	25	25	100
	HSUV201	Creativity, Problem Solving and Innovation	2		2		2	0	0	25	25	50
			15	**16**	**15**	**9**	**24**	**225**	**225**	**150**	**150**	**750**
SY Sem-4	ITUC202	Fundamentals of Database Management Systems	3	4	3	2	5	50	50	25	25	150
	CEUC201	Fundamentals of Software Engineering	4	0	4	0	4	50	50	0	0	100
	CEUC202	Computer Organization and Architecture	3	0	3	0	3	50	50	0	0	100
	ITUE204	Design and Analysis of Algorithms	4	2	4	1	5	50	50	25	25	150
	XXXXX	Elective – I	4	2	4	1	5	50	50	25	25	150
	ITUP201 CEUP201 CSUP201	Software Group Project	0	2	0	2	2	0	0	25	25	50
	XXXXX	University Elective - II	0	4	0	2	2	0	0	25	25	50
	HSUV202 HSUV206	Human Values and Ethics OR Life Lessons from Ramayan and Mahabharat	2		2		2	0	0	25	25	50
			18	**16**	**18**	**10**	**28**	**250**	**250**	**150**	**150**	**800**

CHAROTAR UNIVERSITY OF SCIENCE & TECHNOLOGY (CHARUSAT)

LIST OF ELECTIVE SUBJECTS FOR B TECH PROGRAMME IN CE/CSE/IT

	Code	University Elective – I (Sem-3)	Code	University Elective – II (Sem-4)	Code	Elective – I (Sem-4)
ELECTIVES	MEUE201	Engineering Graphics and Design	MEUE202	Nature And Properties of Materials	CSUE203	Foundation Of Data Science and Analytics
	EEUE201	Fundamentals Of Electrical Engineering	EEUE202	Solar Energy Engineering and Technology	CSUE204	Foundation Of Embedded System
	CLUE201	Environment And Development	CLUE202	Ecology And Environment	CEUE204	Introduction To Computer Graphics
	ECUE201	Scientific Computing Using MATLAB	ECUE202	Introduction To Internet of Things	ITUE205	Fundamentals Of Information Security
	CEUE201	The Joy Of Computing Using Python	CEUE202	Software Conceptual Design		
	ITUE201	Introduction To Quantum Computing	ITUE202	Ethical Hacking		
	CSUE201	Python For Data Science	CSUE202	Google Cloud Computing Foundations		
	AIUE201	An Introduction to Artificial Intelligence	AIUE202	Social Network Analysis		
	BMUD201	Money And Banking	BMUD251	Economics Of Health And Health Care		
	CAUD203	Human Computer Interactions	CAUD204	Modern Application Development		
	PTUD191	Basics Of Health Promotion and Education Intervention	PTUD192	Ergonomics Workplace Analysis		
	NRMD251	First Aid Masterclass- A Complete Guide to First Aid	NRMD261	Mindfulness And Well-Being: Living With Balance And Ease		

B. Tech. (CE/CSE/IT) Programme

SYLLABI

(Semester – 3)

CHAROTAR UNIVERSITY OF SCIENCE AND TECHNOLOGY

Charotar University of Science and Technology (CHARUSAT)
Faculty of Technology and Engineering (FTE)

Subject: Fundamentals of Computer Networks (ITUC201)

Semester: 3rd

Teaching Scheme:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/Week	4	2	-	6	5
Marks	100	50	-	150	

Course Pre-requisites:

- Data Communication Components: https://www.geeksforgeeks.org/components-of-data-communication-system/

Course Description:

The course introduces main concepts of networking, application areas, classification, reference models, transmission environment, technologies, routing algorithms, IP, UDP and TCP protocols, reliable data transferring methods, application protocols, management systems, and perspectives of communication networks. The course structure consists of lectures, tutorials, laboratory works in computer classroom and individual work.

Course Objectives:

1. To gain a comprehensive understanding of networking concepts at different layers of OSI model.
2. To familiarize students with switching and routing technologies.
3. To teach troubleshooting and management of wired networks.
4. To utilize industry-standard tools for creating topologies and realize data transmission across the devices.
5. To work and explore with routing protocols.

Course Outcomes:

By the end of the course, students will be able to:

1. **CO1**: Analyze and identify innovative and feasible solutions using technical and domain knowledge.
2. **CO2**: Apply interdisciplinary skills and tools for software/system development.
3. **CO3**: Demonstrate effective team collaboration, project planning, and time management.
4. **CO4**: Implement, test, and evaluate the system, including performance and security aspects.
5. **CO5**: Prepare structured technical documentation and deliver professional project presentations.

- **Pre-requisites online video/course:**
  - Introduction to N/W & Data Communication by IIT Madras, https://www.youtube.com/watch?v=y3mCx2StKxg .

- **Self-study/Further Study components and materials:**

Unit No.	Unit/Topic Details
1	**INTRODUCTION**
	1.1 Introduction to Computer Networks, types of Networks, Network Applications
	Materials: - ➢ https://book.systemsapproach.org/foundation/applications.html ➢ https://www.youtube.com/watch?v=VwN91x5i25g&list=PLBlnK6fEyqRgneraVKkEXrwyLVx2vJUvt ➢ https://www.geeksforgeeks.org/types-of-computer-networks/
	1.2 Network Hardware
	Materials: - ➢ https://www.spiceworks.com/tech/networking/articles/what-is-network-hardware/ ➢ https://www.youtube.com/watch?v=0pMm_QxCg3I
	1.3 Network Software
	Materials: - ➢ https://www.spiceworks.com/tech/networking/articles/what-is-network-software/ ➢ https://www.youtube.com/watch?v=k8NmM-hImBU
	1.4 Internet
	Materials: - ➢ https://www.cloudflare.com/learning/network-layer/how-does-the-internet-work/ ➢ https://www.youtube.com/watch?v=UXsomnDkntI
	1.5 The Physical Layer: Theoretical Basis for Communication
	Materials: - ➢ https://www.geeksforgeeks.org/physical-layer-in-osi-model/ ➢ https://www.youtube.com/watch?v=rKzDbdGhcdY
2	**THE DATA LINK LAYER AND THE MEDIUM ACCESS SUBLAYER**
	2.1 Sliding Window Protocols
	Materials: - ➢ https://www.javatpoint.com/sliding-window-protocol ➢ https://www.youtube.com/watch?v=5A-nvZHFHX0&t=36s
	2.2 The Data Link Layer in the Internet
	Materials: - ➢ https://www.javatpoint.com/data-link-layer ➢ https://www.youtube.com/watch?v=N1apF49Ih28
	2.3 Multiple Access Protocols
	Materials: - ➢ https://www.geeksforgeeks.org/multiple-access-protocols-in-computer-network/ ➢ https://www.youtube.com/watch?v=YAjfUc7Tt24&list=PLTmU03EUNiEeHQq1chuiRBb94OspHJtsV
	2.4 Wireless LAN
	Materials: - ➢ https://www.youtube.com/watch?v=DAR52r0lEtw ➢ https://www.pynetlabs.com/wireless-lan-wlan/
	2.5 Broadband Wireless
	Materials: - ➢ https://www.techtarget.com/searchmobilecomputing/definition/wireless-broadband ➢ https://www.youtube.com/watch?v=54tBzrDUJd8

3	**THE NETWORK LAYER**
	3.1 Network Layer Design Issues
	Materials: - ➢ https://www.tutorialspoint.com/network-layer-design-issues
	3.2 Routing Algorithms
	Materials: - ➢ https://www.geeksforgeeks.org/classification-of-routing-algorithms/ ➢ https://www.youtube.com/watch?v=1KGC7Tp6HGo
4	**THE TRANSPORT LAYER**
	4.1 Simple Transport Protocol
	Materials: - ➢ https://www.tutorialspoint.com/what-are-the-elements-of-transport-protocol ➢ https://www.youtube.com/watch?v=x1HH6khHrBA
5	**THE APPLICATION LAYER**
	5.1 Domain Name System
	Materials: - ➢ https://www.cloudflare.com/learning/dns/what-is-dns/
	5.2 Electronic Mail
	Materials: - ➢ https://www.sciencedirect.com/topics/social-sciences/electronic-mail ➢ https://www.youtube.com/watch?v=ELioaGg4WOY
	5.3 World Wide Web: Architectural Overview
	Materials: - ➢ https://www.studytonight.com/computer-networks/world-wide-web ➢ https://www.youtube.com/watch?v=raywYV-dKSQ

Syllabus:

Unit No.	Unit/Topic Details		Hours (hr)	Evaluation Weightage (%)
1	**INTRODUCTION**		05	10
	1.1	Reference Models: OSI, TCP/IP		
	1.2	Connection-Oriented Network: X.25, Frame Relay		
	1.3	Guided Transmission Media		
	1.4	Wireless Transmission		
	1.5	The Public Switched Telephone Networks		
	1.6	Mobile Telephone System		
2	**THE DATA LINK LAYER AND THE MEDIUM ACCESS SUBLAYER**		17	30
	2.1	Design Issues		
	2.2	Error Detection and Correction		
	2.3	Elementary Data Link Protocols		
	2.4	Example Data Link Protocols: HDLC		
	2.5	Channel Allocation Problem		
	2.6	Ethernet Overview		
	2.7	Data Link Layer Switching		
	2.8	Wireless LAN 802.11a/b/g/n Overview		
	2.9	COAP Protocol Overview		

3	**THE NETWORK LAYER**		15	25
	3.1	Dynamic and Static routing Algorithm Vs Distance Vector and Link State Routing Algorithm		
	3.2	Congestion Control Algorithms		
	3.3	Internetworking		
	3.4	The Network Layer in the Internet (IPv4 and IPv6)		
	3.5	Quality of Service		
4	**THE TRANSPORT LAYER**		10	15
	4.1	Transport Service		
	4.2	Elements of Transport Protocol		
	4.3	Internet Transport Layer Protocols: UDP and TCP		
5	**THE APPLICATION LAYER**		13	20
	5.1	Application Layer Protocols		
	5.2	Network Management and Simple Network Management Protocol (SNMP)		
	5.3	File Transfer Protocol (FTP)		
	5.4	Hypertext Transfer Protocol (HTTP/HTTPS)		
	5.5	Simple Mail Transfer Protocol (SMTP)		
	5.6	TELNET		
		Total Hours:	**60**	

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	-	-	-	-	-	-	-	-	-	-	3	-
CO2	-	3	3	-	2	-	-	-	-	-	-	-	3	-
CO3	-	-	-	3	2	-	-	-	-	-	-	-	2	-
CO4	-	-	-	-	3	-	-	-	2	-	-	-	-	3
CO5	-	3	3	-	-	-	-	-	-	-	-	2	3	-

The correlation levels 1, 2 or 3 are as defined below:
1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High), If there is no correlation, put "-" placed.

Recommended Study Material:

❖ **Text books:**

1. "Computer Networks", Fifth Edition by David J. Wetherall, Andrew S. Tanenbaum Released September 2010 Publisher(s): Pearson ISBN: 9780133485936.
2. "Data Communications and Networking", 4th Edition Behrouz A. Forouzan ISBN: 9780073250328 Publication Date & Copyright: 2007. The McGraw-Hill Companies, Inc.

❖ **Reference Books:**

1. "Internetworking with TCP/IP Volume One" by Douglas E. Comer for A deep dive into TCP/IP architecture and protocols.
2. "High-Performance Communication Networks" by Jean Walrand and Pravin Varaiya Ideal for understanding advanced concepts like quality of service (QoS).
3. "Communication Networks" by Alberto Leon-Garcia and Indra Widjaja which Covers theoretical aspects with a focus on network performance analysis.

4. "Wireless Communications and Networking" by Vijay Garg for the Detailed coverage of wireless LANs, mobile telephony, and broadband wireless.
5. "Routing TCP/IP, Volume I & II" by Jeff Doyle and Jennifer DeHaven Carroll for Advanced guide to routing concepts and protocols.

Online Resources:

1. **Introduction to Computer Networking Basics (Simplilearn)**
   - Focuses on design principles and networking protocols.
2. **Computer Networking - Wired and Wireless Networks and Protocols (Alison)**
   - Focuses on networking protocols.
3. **Computer Networking - Local Area Networks and the OSI Model (Alison)**
   - Local Area Networks (LAN) and the seven layers of the networking OSI Model.
4. **Advanced Networking by Georgia Tech (Udacity)**
   - Explores advanced networking concepts like congestion control and dynamic routing.

Practical/Lab Work Implementation:

- **Practical Tools:** CISCO Packet Tracer, NS2, NS3
- Lab work should be based on performance analysis of the different types of topologies and different protocols (not limited to) at TCP/IP layers using Open Source Network simulators/propriety network simulators like Packet Tracer, NS2, and NS3 etc.
- Lab work should be also based on implementation of Client, Server and peer to peer Communication with open-source architectures.
- At least 1-2 problem definition should be based on COAP Protocol.
- Network disaster management/ Network troubleshooting, and Network management-based problem definition should be there.
- Problem definition/definitions should be based on Routing algorithm testing, execution, performance analysis.

Care Taking Points:

- Self-study topics are part of evaluation and clarity about this should be provided to the students in the first session.
- Practical list should be based on Practical/Lab Work Implementation guidelines.

Charotar University of Science and Technology (CHARUSAT)
Faculty of Technology and Engineering (FTE)

Subject: Fundamentals of Data Structure and Algorithms (CSUC201)

Semester: 3rd

Teaching Scheme:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/Week	3	2	-	5	4
Marks	100	50	-	150	

Course Pre-requisites:

- Programming Language Concepts (C or C++ or Java)

Course Description:

This course delves into essential data structures and algorithms, focusing on their implementation, applications, and performance analysis. Students will gain expertise in solving computational problems using arrays, linked lists, stacks, queues, trees, graphs, sorting, and searching algorithms. Advanced topics like dynamic programming, greedy algorithms, and higher-dimensional data structures are also explored, with practical applications in domains such as geospatial systems, compression, and scheduling.

Course Objectives:

1. To understand Fundamental Concepts of data structures (such as arrays, linked lists, stacks, queues, trees, and graphs) and algorithms, including their characteristics, uses, and performance.
2. To develop skills to analyze and compare the efficiency of algorithms using Big O notation, focusing on time and space complexity.
3. Learn to implement various data structures in a programming language of choice, ensuring a practical understanding of their functionalities and applications.
4. To enhance problem-solving abilities by tackling a variety of algorithmic challenges and real-world applications.

Course Outcomes:

By the end of the course, students will be able to:

1. **CO1**: Understand different types of data structures, analyze upper bound of algorithms, and represent data structures in memory.
2. **CO2**: Implement linear and non-linear data structures, along with understanding their practical applications.
3. **CO3**: Perform searching and sorting efficiently, understand upper bound of sorting algorithms, and apply different sorting algorithms to real-world problems.
4. **CO4**: Implement and analyze hash tables, understand their significance, and use them in real-world applications such as dictionaries.

5. **CO5**: Develop the skills to select suitable data structures for addressing complex computational challenges.

- **Pre-requisites online video/ courses:**
  - Introduction to Data Structures and Algorithms by IIT Delhi, https://nptel.ac.in/courses/106102064 .
  - Introduction to Graph Algorithms by IISc Bangalore, https://onlinecourses.nptel.ac.in/noc24_cs70/preview .

- **Self-Study/Further Study components and materials:**

Unit No.	Unit/Topic Details	
1	**INTRODUCTION TO DATA STRUCTURES AND ALGORITHMS**	
	1.1	Study iterative and recursive problems like Factorial, Fibonacci series, Euclidean GCD algorithm.
	1.2	Explore real-world applications of algorithmic complexity (e.g., performance bottlenecks).
	Materials: ➢ Introduction to Data Structures and Algorithms • https://www.geeksforgeeks.org/introduction-to-data-structures/ ➢ Big-O Notation – GeeksforGeeks • https://www.geeksforgeeks.org/analysis-algorithms-big-o-analysis/	
2	**ARRAYS AND LINKED LISTS**	
	2.1	Implement and analyze image representation in memory using array.
	2.2	Explore the role of linked lists in implementing undo and redo concepts in text editor.
	Materials: ➢ Arrays and Linked Lists • https://www.geeksforgeeks.org/array-data-structure-guide/ • https://www.geeksforgeeks.org/linked-list-data-structure/	
3	**SORTING AND SEARCHING**	
	3.1	Explore applications of sorting in databases and e-commerce (e.g., ranking).
	3.2	Practice implementing counting and radix sorts for large datasets.
	3.3	Explore applications of Binary search techniques to solve real-world applications.
	Materials: ➢ Linear Search – GeeksforGeeks • https://www.geeksforgeeks.org/linear-search/ ➢ Binary Search – GeeksforGeeks • https://www.geeksforgeeks.org/binary-search/	
4	**STACKS AND QUEUES**	
	4.1	Implement a browser history system using stacks.
	4.2	Explore real-world queue applications in job scheduling and simulation.
	Materials: ➢ Stacks and Queues • https://www.geeksforgeeks.org/stack-data-structure/ • https://www.geeksforgeeks.org/queue-data-structure/	
5	**TREES AND BINARY SEARCH TREES**	

	5.1	Practice tree traversal techniques in real-world hierarchical data (e.g., XML parsers).
	5.2	Implement balancing techniques in AVL and Red-Black Trees.
	Materials: ➢ Tree Traversals • https://www.geeksforgeeks.org/tree-traversals-inorder-preorder-and-postorder/ ➢ Binary Search Tree • https://www.geeksforgeeks.org/binary-search-tree-data-structure/ ➢ AVL Trees – GeeksforGeeks • https://www.geeksforgeeks.org/introduction-to-avl-tree/ ➢ **Heap Tree** • https://www.geeksforgeeks.org/heap-data-structure/	
6	**GRAPHS**	
	6.1	Implement graph-based solutions for social networks (e.g., friend recommendations).
	6.2	Explore applications of MST in network design and logistics.
7	**HASHING**	
	7.1	Explore hashing techniques in database indexing
	7.2	Explore hashing in cryptography for digital signature , message authentication.
	Materials: ➢ Hashing in Data Structure – GeeksforGeeks • https://www.geeksforgeeks.org/hashing-data-structure/	
8	**HIGHER DIMENSIONAL DATA STRUCTURES**	
	8.1	Implement spatial data queries using K-d Trees.
	8.2	Explore Quad Trees in image compression and Interval Trees in scheduling.

Syllabus:

Unit No.	Unit/Topic Details	Hours (hr)	Evaluation Weightage (%)
1	**INTRODUCTION TO DATA STRUCTURES AND ALGORITHMS** 1.1 Introduction to Data Structure, Types of data structure, Data Types, Introduction to Algorithms. 1.2 Big-O, Big-Ω, Big-Θ notations 1.3 Recursive algorithms and their complexity 1.4 Time and space complexity analysis **Prerequisites:** ➢ Understanding of basic programming constructs (loops, functions). ➢ Familiarity with iterative algorithm and basic mathematical concepts (e.g., summation, induction). **Materials:** ➢ Introduction to Data Structures and Algorithms • https://www.geeksforgeeks.org/introduction-to-data-structures/ ➢ Big-O Notation – GeeksforGeeks • https://www.geeksforgeeks.org/analysis-algorithms-big-o-analysis/ ➢ Data Structures and Algorithms • https://www.codechef.com/roadmap/data-structures-and-algorithms	03	05

2	**ARRAYS AND LINKED LISTS**		06	15
	2.1	Arrays: Memory Representation of Array in Row Major Order and Column Major Order (1-Dimensional, 2-Dimensionall, 3-Dimensional, Multi-Dimensional), Applications of Array		
	2.2	Linked lists: Memory Representation of Linked List, Types of Linked List: Singly Linked List, Singly Linked List Operations (Insert, Delete, Search, Display, Reverse), Doubly Linked List Operations (Insert, Delete, Search, Display, Reverse), Circular Singly Linked List, Circular Doubly Linked List, Applications of Linked List		
	Prerequisites: ➢ Understanding of array indexing and basic data storage concepts. ➢ Familiarity with pointers or references in programming. **Materials:** ➢ Arrays and Linked Lists • https://www.geeksforgeeks.org/array-data-structure-guide/ • https://www.geeksforgeeks.org/linked-list-data-structure/			
3	**SORTING AND SEARCHING**		05	12
	3.1	Searching Algorithms: Linear and Binary Search (Iterative and Recursive)		
	3.2	Sorting Algorithms: Selection Sort, Bubble Sort, Insertion Sort, Counting sort, Radix sort		
	3.3	Time complexity analysis		
	Prerequisites: ➢ Familiarity with arrays and basic iteration constructs. ➢ Understanding of iterative and recursion algorithms. **Materials:** ➢ Sorting Visualization • https://www.cs.usfca.edu/~galles/visualization/ComparisonSort.html ➢ Binary Search - GeeksforGeeks • https://www.geeksforgeeks.org/binary-search/			
4	**STACKS AND QUEUES**		06	18
	4.1	Stacks: Memory Representation of Stack using Array and Linked List, Operations: push, pop, peep, change, Performance Analysis of operations		
	4.2	Applications of Stack: • Recursion: Head Recursion and Tail Recursion, Tower of Hanoi Problem • Conversion: Infix to Postfix, Infix to Prefix • Evaluation: Prefix and Postfix expression		
	4.3	Queues: Memory Representation of queue using Array and Linked List, Simple Queue: Insert and Delete operation, Circular Queue: Insert and Delete operation, Performance Analysis of operations on Queue,		
	4.4	Double-ended Queue, Applications of Queue		
	Prerequisites: ➢ Understanding of arrays and linked lists. ➢ Familiarity with problem-solving techniques.			

<table>
<tr><td></td><td colspan="2">Materials:
➢ Stacks and Queues
• https://www.geeksforgeeks.org/stack-data-structure/
• https://www.geeksforgeeks.org/queue-data-structure/
➢ Circular Queue
• https://www.geeksforgeeks.org/introduction-to-circular-queue/
➢ Expression Evaluation Using Stack
• https://www.geeksforgeeks.org/videos/evaluation-of-postfix-expression-dsa-problem/</td><td></td><td></td></tr>
<tr><td rowspan="8">5</td><td colspan="2">TREES AND BINARY SEARCH TREES</td><td rowspan="8">12</td><td rowspan="8">25</td></tr>
<tr><td>5.1</td><td>Tree Concepts, Memory Representation of Tree, Applications of Tree, Binary Tree and types, Tree Traversal Techniques: Pre-order, Post order and In-order</td></tr>
<tr><td>5.2</td><td>Binary Search Tree: Insert, Delete and Search Operations with analysis</td></tr>
<tr><td>5.3</td><td>AVL Trees: Insert, Delete and Search Operations with analysis</td></tr>
<tr><td>5.4</td><td>Tries, Red black tree, Multi-way search tree</td></tr>
<tr><td>5.5</td><td>Heaps: Binary Heaps, heap sort, Priority Queue</td></tr>
<tr><td colspan="2">Prerequisites:
➢ Understanding recursion and binary relationships.
➢ Familiarity with hierarchical data structures.
Materials:
➢ Trees Traversals
• https://www.geeksforgeeks.org/tree-traversals-inorder-preorder-and-postorder/
➢ Binary Search Tree
• https://www.geeksforgeeks.org/binary-search-tree-data-structure/
➢ AVL Trees – GeeksforGeeks
• https://www.geeksforgeeks.org/introduction-to-avl-tree/
➢ Heap Tree
• https://www.geeksforgeeks.org/heap-data-structure/</td></tr>
<tr><td rowspan="5">6</td><td colspan="2">GRAPHS</td><td rowspan="5">5</td><td rowspan="5">15</td></tr>
<tr><td>6.1</td><td>Graph concepts, Memory Representation of Graph</td></tr>
<tr><td>6.2</td><td>Graph Traversals: Breadth First Search (BFS) and Depth First Search (DFS)</td></tr>
<tr><td>6.3</td><td>Topological sort, Applications of Graph</td></tr>
<tr><td colspan="2">Prerequisites:
➢ Understanding of Array and Linked List theory concepts.
Materials:
➢ DFS and BFS
• https://www.geeksforgeeks.org/breadth-first-search-or-bfs-for-a-graph/
• https://www.geeksforgeeks.org/depth-first-search-or-dfs-for-a-graph/</td></tr>
<tr><td rowspan="4">7</td><td colspan="2">HASHING</td><td rowspan="4">04</td><td rowspan="4">07</td></tr>
<tr><td>7.1</td><td>Introduction of Hashing, Applications of Hashing</td></tr>
<tr><td>7.2</td><td>Direct Address Table, Hashing Functions, properties of hash functions</td></tr>
<tr><td>7.3</td><td>Collision-Resolution Techniques- open addressing and chaining</td></tr>
</table>

	Prerequisites: ➢ Familiarity with recursion and divide-and-conquer algorithms. ➢ Understanding of basic optimization problems. **Materials:** ➢ Hashing in Data Structure – GeeksforGeeks • https://www.geeksforgeeks.org/hashing-data-structure/		
8	**HIGHER DIMENSIONAL DATA STRUCTURES** **8.1** Sparse matrices **8.2** K-d Trees, Quad Trees, Interval Trees **Prerequisites:** ➢ Understanding of arrays and matrix operations. ➢ Familiarity with recursive partitioning techniques. **Materials:** ➢ Sparse matrices • https://www.geeksforgeeks.org/sparse-matrix-representation/ ➢ K-d Trees – GeeksforGeeks • https://www.geeksforgeeks.org/search-and-insertion-in-k-dimensional-tree/ ➢ Interval Trees – GeeksforGeeks • https://www.geeksforgeeks.org/interval-tree/	04	03
	Total Hours:	**45**	

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3	2	2	2	-	-	-	-	-	-	2	3	2
CO2	3	3	3	2	2	-	-	-	-	-	-	2	3	3
CO3	3	3	3	3	2	-	-	-	-	-	-	2	3	3
CO4	3	3	2	3	3	-	-	-	-	-	-	2	3	3
CO5	3	3	3	3	2	-	-	-	-	-	-	3	3	3

The correlation levels 1, 2 or 3 are as defined below:
1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High), If there is no correlation, put "-" placed.

Recommended Study Material:

❖ **Text books:**

1. "Data Structures using C & C++", by Tanenbaum, Prentice-Hall International.
2. "An Introduction to Data Structures with Applications", by Jean-Paul Tremblay & Paul G. Sorenson, Tata McGraw Hill.

❖ **Reference Books:**

1. "Fundamentals of Data Structures in C++", by Sartaj Sahani, Galgotia.Publishers.
2. "Introduction to Algorithms", by Cormen, Leiserson, Rivest, and Stein (CLRS), 3rd Edition.
3. "Algorithms", by Robert Sedgewick.
4. "Data Structures and Algorithm Analysis in C", by Mark Allen Weiss.

Charotar University of Science and Technology (CHARUSAT)
Faculty of Technology and Engineering (FTE)

Subject: Discrete Mathematics (MSUD203)

Semester: 3rd

Teaching Scheme:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/Week	3	-	-	3	3
Marks	100	-	-	100	

Course Pre-requisites:

- Understanding of basic Set Theory and Matrices, Familiar with Elementary Calculus.

Course Description:

This course provides a foundational understanding of discrete mathematical structures and linear algebra, emphasizing their applications in computer science, engineering, and problem-solving. Key topics include mathematical logic, relations and lattices, graph theory, group theory, and vector spaces. The course integrates theoretical concepts with practical applications, preparing students for advanced studies and professional challenges in technology and engineering.

Course Objectives:

1. Develop logical reasoning skills through the study of mathematical logic and proof techniques.
2. Understand and analyze relations, lattices, and their applications in discrete structures.
3. Explore fundamental concepts in graph theory and their significance in computational problems.
4. Provide students with an understanding of the basic algebraic structures such as groups, groupoids, semigroups, monoids, and Abelian groups, and their properties.
5. Apply linear algebra concepts, such as vector spaces and linear transformations, to solve engineering and scientific problems.

Course Outcomes:

By the end of the course, students will be able to:

1. **CO1**: Construct and evaluate logical arguments using propositional and predicate calculus.
2. **CO2**: Explain the basic properties of relations, represent relations graphically and using matrices, and analyze their characteristics.
3. **CO3**: Solve problems using graph-theoretical concepts, including paths, circuits, and graph coloring.
4. **CO4**: Define and explain the basic properties of groups, groupoids, semigroups, monoids, and Abelian groups, and distinguish between these structures.
5. **CO5**: Demonstrate proficiency in linear algebra techniques, including vector space analysis and linear transformations.

- **Pre-requisites online video / course:**
  - Basic calculus for Engineers, Scientists and economists, https://archive.nptel.ac.in/courses/111/104/111104085/#

Syllabus:

Unit No.	Unit/Topic Details		Hours (hr)	Evaluation Weightage (%)
1	**MATHEMATICAL LOGIC**		07	11
	1.1	Induction, Propositions, Combination of Propositions, Logical Operators & Propositional		
	1.2	Algebra, Equivalence, Predicates & Quantifiers, Interaction of Quantifiers with Logical Operators		
	1.3	Logical Interference & Proof Technique		
2	**RELATIONS AND LATTICE:**		07	15
	2.1	Definition & Basic Properties, Graphs Of Relation, Matrices Of Relation		
	2.2	Equivalence Relation, Equivalence Classes, Partition		
	2.3	Partial Ordered Relation, Posets, Hasse Diagram, Upper Bounds, Lower Bound		
	2.4	GLB & LUB of Sets, Definition & Properties of Lattice		
	2.5	Sub Lattice, Distributive & Modular Lattices, Complemented & Bounded Lattices, Complete Lattices & Boolean Algebra		
3	**GRAPH THEORY:**		10	23
	3.1	Graph Definition, Graph Representation		
	3.2	Basic Concepts Of Finite & Infinite Graph, Incidence & Degree, Isomorphism		
	3.3	Subgraph, Walk, Path & Circuits, Cliques, Cycles and Loops		
	3.4	Operations On Graphs, Connected Graph, Disconnected Graph & Components, Eulerian and Hamiltonian graph		
	3.5	Complete Graph, Regular Graph, Bipartite Graph, Planar Graphs, Subgraphs, spanning subgraphs, isomorphic graphs		
	3.6	Weighted Graphs, Directed & Undirected Graphs, Connectivity of Graphs		
	3.7	Matching in graphs, Graph coloring		
4	**GROUP THEORY:**		10	18
	4.1	Basic Properties of Group, Groupoid, Semigroup & Monoid, Abelian Group		
	4.2	Subgroup, Cosets, Normal Subgroup, Lagrange's Theorem, Cyclic Group		
	4.3	Permutation Group, Homomorphism & Isomorphism of Groups		
5	**LINEAR ALGEBRA:**		11	33
	5.1	Vector space: definition and examples, Subspaces		
	5.2	Linear combinations, linearly dependence and linearly independence		
	5.3	Basis and dimension of a vector space		

	5.4	Linear transformations, Null space and range of a linear transformation, Rank - nullity theorem, Isomorphisms		
		Total Hours:	**45**	

- **Post Self-study components and materials:**

Unit No.	Unit/Topic Details	
1	**MATHEMATICAL LOGIC:**	
	1.1	Study additional proof techniques, including contradiction and induction.
	1.2	Explore the applications of propositional logic in digital circuits.
2	**RELATIONS AND LATTICE:**	
	2.1	Analyze real-world examples of equivalence relations and partitions.
	2.2	Explore the use of Hasse diagrams to visualize partial orders.
3	**GRAPH THEORY:**	
	3.1	Study algorithms for finding Eulerian paths, Hamiltonian circuits, and Graph Coloring.
	3.2	Applications of graph theory in Network Design and Social Networks.
4	**GROUP THEORY:**	
	4.1	Explore practical applications of groups in cryptography.
	4.2	Explore the use of Hasse diagrams to visualize partial orders.
5	**LINEAR ALGEBRA:**	
	5.1	Explore real-world applications of vector spaces, such as in Computer Graphics and Machine Learning.
	5.2	Study Singular Value Decomposition (SVD) and its application in recommendation systems.

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	2	3	2	1	-	-	-	-	-	-	-	-
CO2	3	3	2	2	2	1	-	-	-	-	-	-	-	-
CO3	3	3	3	2	2	1	-	-	-	-	-	-	-	-
CO4	3	2	2	2	1	1	-	-	-	-	-	-	-	-
CO5	3	3	3	3	3	1	-	-	-	-	-	-	-	-

The correlation levels 1, 2 or 3 are as defined below:
1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High), If there is no correlation, put "-" placed.

Recommended Study Material:

- **Text books:**

1. Rosen, Kenneth H. and Kamala Krithivasan, "Discrete mathematics and its applications", Vol. 6. New York: McGraw-Hill, 1995.
2. Swapan Kumar Sarkar, "A Text Book of Discrete Mathematics", S. Chand and Company, New Delhi, 2008.
3. H. Anton and C. Rorres, "Elementary Linear Algebra, Application version", Wiley Edition 2010.
4. Tremblay, Jean Paul, and Rampurkar Manohar, "Discrete mathematical structures with applications to computer science", McGraw-Hill, Inc., 1975.

5. Lay, David C, “Linear algebra and its applications”, Pearson Education India, 2003.

❖ **Reference Books:**

1. D. F. McAllister and D. F. Stanat, “Discrete Mathematics in Computer Science”, Prentice-Hall, Inc.1977.
2. Narsingh Deo, “Graph theory with applications to engineering and computer science”, Courier Dover Publications, 2016.
3. B. Kolman and R. C. Busby, “Discrete Mathematical Structures for Computer Science”, 2nd edition, Prentice-Hall, Englewood Cliffs, New Jersey 1987.
4. D. S. Malik and Mridul K. Sen, “Discrete mathematical structures: theory and applications”, Course Technology, 2004.
5. H. Cormen Thomas, C. E. Leiserson, R. L. Rivest and C. Stein, “Introduction to algorithms”, (Vol. 6). Cambridge: MIT press,2001.
6. Liu, Chung Laung, “Elements of discrete mathematics”, McGraw-Hill, 1985.

Online Resources:

1. NPTEL Course – Discrete Mathematics
   Link: - https://onlinecourses.nptel.ac.in/noc20_cs82/preview

2. Coursera, Introduction to Discrete Mathematics for Computer Science Specialization
   Link: - https://www.coursera.org/specializations/discrete-mathematics

3. Blue1Brown: Essence of Linear Algebra
   Link: -
   https://www.youtube.com/playlist?list=PLZHQObOWTQDPD3MizzM2xVFitgF8hE_ab

Charotar University of Science and Technology (CHARUSAT)
Faculty of Technology and Engineering (FTE)

Subject: Web Development Frameworks (ITUE203)

Semester: 3rd

Teaching Scheme:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/Week	3	4	-	7	5
Marks	100	50	-	150	

Course Pre-requisites:

- Overview of Programming Concepts
- Understanding of basic Hypertext Markup Language elements (headings, paragraphs, links, images) and Cascading Style Sheets styling (colors, fonts, and layouts).

Course Description:

This course will cover the basics of the Internet, building a web application, databases, performance and security, and building a web application. In addition, the course will have an extensive set of Practical Tutorials which will help students get a feel for real-world development.

Course Objectives:

1. Equip students with a strong foundation in web development by teaching the core concepts of Hypertext Markup Language, Cascading Style Sheets, and JavaScript to create responsive and interactive web pages.
2. Learn to create robust and scalable back-end systems using PHP, a widely-used server-side scripting language.
3. Integrate front-end and back-end components to build complete, data-driven web applications.
4. Apply best practices for security, including input validation, password hashing, and protection against SQL injection.

Course Outcomes:

By the end of the course, students will be able to:

1. **CO1**: Design and develop responsive and interactive web application using Hypertext Markup Language elements, forms, multimedia, and layouts.
2. **CO2**: Demonstrate Cascading Style Sheets styling techniques such as selectors, text formatting, page formatting, and box model concepts to enhance the presentation of web pages.
3. **CO3**: Implement dynamic behaviours and user interactions using JavaScript (ES6) features, including conditional branching, loops, functions, and operators.
4. **CO4**: Manipulate the Document Object Model to create, update, or remove elements and add interactivity to web applications through event handling.
5. **CO5**: Demonstrate server-side functionality, including handling forms, managing sessions and cookies, and implementing Object-Oriented Programming principles.

- **Self-study/Further Study components and materials:**

Unit No.	Unit/Topic Details	
1	**INTRODUCTION TO WEB TECHNOLOGIES**	
	1.1	Terminologies: Web client, Web server, Web browser, Webpage, Website, Client-server architecture
	Materials: ➢ https://www.geeksforgeeks.org/what-is-a-web-server-working-and-architecture/	
2	**BUILDING WEB APPLICATION USER INTERFACE WITH HYPERTEXT MARKUP LANGUAGE 5 AND CASCADING STYLE SHEETS 3**	
	2.1	Hypertext Markup Language Introduction
	2.2	Hypertext Markup Language Documents
	Materials: ➢ https://www.w3schools.com/html/html_basic.asp	
	2.3	Cascading Style Sheets Introduction
	Materials: ➢ https://www.w3schools.com/css/css_intro.asp	
3	**BUILD INTERACTIVE USER INTERFACE WITH JAVASCRIPT FUNDAMENTS (ES6)**	
	3.1	An Introduction to JavaScript
	3.2	Code editors
	3.3	Developer console
	3.4	Variable and Data Types
	3.5	Conditional branching and loops and iteration
	Materials: ➢ https://javascript.info/intro ➢ https://javascript.info/code-editors ➢ https://javascript.info/devtools ➢ https://javascript.info/variables ➢ https://javascript.info/types ➢ https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Loops_and_iteration	
4	**JAVASCRIPT OBJECTS**	
	4.1	Objects
	Materials: ➢ https://javascript.info/object	
5	**BUILD WEB APPLICATION WITH SERVER-SIDE LANGUAGE - PHP**	
	5.1	Introduction to PHP, Variables & Data Types
	5.2	Decisions and loops Structure
	Materials: ➢ https://www.tutorialspoint.com/php/php_introduction.htm ➢ https://www.tutorialspoint.com/php/php_variable_types.htm ➢ https://www.tutorialspoint.com/php/php_data_types.htm ➢ https://www.tutorialspoint.com/php/php_strings.htm ➢ https://www.tutorialspoint.com/php/php_loop_types.htm	

Syllabus:

Unit No.	Unit/Topic Details		Hours (hr)	Evaluation Weightage (%)
1	**INTRODUCTION TO WEB TECHNOLOGIES**		04	6
	1.1	Web Protocols, secure connections, application and development tools		
	1.2	Web site design principles, Planning the site and navigation		
	1.3	Web Application Architecture and it's type, choose the right web application type & architecture		
	1.4	Global server and local server		
2	**BUILDING USER INTERFACE OF WEB APPLICATION**		08	14
	2.1	User-centered design approach		
	2.2	Color theory and psychology		
	2.3	Typography fundamentals (e.g. Elements, Text, and links)		
	2.4	Layout and composition techniques (Tables and layouts, Lists)		
	2.5	Images and iframe, Entities, Forms		
	2.6	Multimedia, and graphics on a webpage		
	2.7	Properties and its value, Selectors, Text formatting		
	2.8	Page formatting (background, absolute, relative, and fixed layout)		
	2.9	Box Model		
3	**BUILD INTERACTIVE USER INTERFACE OF WEB APPLICATION**		06	10
	3.1	Interaction: alert, prompt, confirm		
	3.2	Type Conversions, operators, maths, Comparisons		
	3.4	Functions, Function expressions, Arrow functions		
4	**DOCUMENT OBJECT MODEL (DOM)**		06	10
	4.1	Introduction of Document Object Model, Document Object Model over the JavaScript		
	4.2	Access document element and style		
	4.3	Event listener operations and event handler to an element		
5	**JAVASCRIPT OBJECTS**		10	16
	5.1	Object references, and copying, Garbage collection		
	5.2	Object methods, "this", Constructor, operator "new"		
	5.3	Object to primitive conversion, Methods of primitives		
	5.4	Prescriptive Analytics - Optimization Techniques, Simulation Modelling		
	5.5	Numbers, Strings, Arrays, Array methods		
	5.6	Iterables, JSON methods, toJSON		
6	**BUILD WEB APPLICATION WITH SERVER-SIDE LANGUAGE**		10	16
	6.1	Function		
	6.2	Array		
	6.3	Handling Form with user interface		
	6.4	Session and Cookie		

	6.5	Working with file and Directories		
	6.6	Object Oriented Programming (OOPs) Concept: Classes, Object, Properties, Constructor and destructor		
	6.7	Advance Object-Oriented Programming (OOPs) Concept: Interface, Abstract Class, Object cloning		
7	**CONNECT WEB APPLICATION DATABASE**		08	14
	7.1	Overview of Database		
	7.2	Types of Databases: Relational, NoSQL, Key-Value Stores, Document-based		
	7.3	Understanding Tables, Rows, Columns, and Relationships		
	7.4	Data Manipulation and Data Definition Queries		
	7.5	Where Clause, JOINS, Clone Tables, Temporary Tables		
	7.6	Database Connectivity in Web Applications		
8	**WEB APPLICATION SECURITY**		08	14
	8.1	Authentication with Hypertext Transfer Protocol		
	8.2	Understanding security, and some best practices for web apps		
	8.3	Authentication and sessions		
		Total Hours:	**60**	

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	3	2	3	-	-	-	1	3	-	-	-	-
CO2	3	2	3	2	3	-	-	-	1	3	-	-	-	-
CO3	3	3	3	2	3	-	-	-	1	3	-	-	-	-
CO4	3	3	3	2	3	-	-	-	1	3	-	-	-	-
CO5	3	3	3	3	3	-	-	-	1	3	2	3	-	-

The correlation levels 1, 2 or 3 are as defined below:
1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High), If there is no correlation, put "-" placed.

Recommended coding Languages and tools:

1. Hypertext Markup Language 5 (HML5)
2. Cascading Style Sheets 3(CSS3)
3. MySQL and MongoDB

Recommended Study Material:

❖ **Text books:**

1. "HTML and CSS: Design and Build Websites", (1st Edition) by Jon Duckett.
2. "JavaScript and jQuery: Interactive Front-End Web Development", (1st Edition) by Jon Duckett.
3. "Eloquent JavaScript: A Modern Introduction to Programming", (3rd Edition) by Marijn Haverbeke.
4. "JavaScript: The Definitive Guide", (7th Edition) by David Flanagan.

❖ **Reference books:**

1. "Learning Web Design: A Beginner's Guide to HTML, CSS, JavaScript, and Web Graphics", (5th Edition) by Jennifer Robbins.
2. "You Don't Know JS", Series (6 books) by Kyle Simpson.
3. "Modern JavaScript for the Impatient", by Cay S. Horstmann.
4. "DOM Scripting: Web Design with JavaScript and the Document Object Model", (1st Edition) Jeremy Keith.
5. "Web Development with Node and Express: Leveraging the JavaScript Stack", (6th Edition) by Ethan Brown.

❖ **Additional Online Resources:**

1. MDN Web Docs (Mozilla) URL:
   - **HTML**: https://developer.mozilla.org/en-US/docs/Web/HTML
   - **CSS**: https://developer.mozilla.org/en-US/docs/Web/CSS
   - **Javascript**: https://developer.mozilla.org/en-US/docs/Web/JavaScript
2. JavaScript Info URL:
   - **Javascript:** https://javascript.info/

❖ **Online Courses**

1. IBM Full Stack Software Developer
   Link: https://www.coursera.org/professional-certificates/ibm-full-stack-cloud-developer
2. Web Development Fundamentals (IBM Skillsbuild)
   Link: https://students.yourlearning.ibm.com/activity/PLAN-43A030B97485?_gl=1*7aja72*_ga*MTIxMzMyMzkzMC4xNzQyMTg3OTIx*_ga_FYECCCS21D*MTc0MjE4NzkyMS4xLjAuMTc0MjE4NzkyMS4wLjAuMA..&utm_source=skillsbuild.org
3. HTML, CSS, and Javascript for Web Developers Specialization
   Link: https://www.coursera.org/specializations/html-css-javascript-for-web-developers

Charotar University of Science and Technology (CHARUSAT)
Faculty of Technology and Engineering (FTE)

Subject: Object Oriented Programming (CEUE203)

Semester: 3rd

Teaching Scheme:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/Week	2	2	-	4	3
Marks	50	50	-	100	

Course Pre-requisites:

- Basic understanding of programming concepts and familiarity with Object-Oriented Programming (OOP) principles.

Course Description:

This course provides a comprehensive introduction to the Java programming language, equipping students with a solid foundation in both its theoretical concepts and practical applications. It begins with fundamental topics such as Java syntax, data types, and control structures. The curriculum then delves into object-oriented programming principles, including classes, objects, inheritance, and polymorphism. Students will explore key Java features like exception handling, the Collections Framework, and input/output operations. The course also introduces advanced concepts such as multithreading and concurrency, ensuring participants gain the skills necessary for developing robust Java applications.

Course Objectives:

1. To understand the fundamentals of Java programming, including syntax, data types, and control structures, to build a strong programming foundation.
2. To learn the principles of Object-Oriented Programming (OOP) through classes, inheritance, interfaces, and packages, enabling modular and reusable code design.
3. To develop proficiency in handling arrays and strings for efficient data manipulation and processing in Java applications.
4. To explore advanced concepts such as multithreaded programming, exception handling, and file I/O operations, ensuring robust and concurrent application development.
5. To master the Java Collections Framework and generics for effective data management and implementation of reusable, type-safe code structures.

Course Outcomes:

By the end of the course, students will be able to:

1. **CO1**: Demonstrate a comprehensive understanding of Java fundamentals, including syntax, data types, control structures, and object-oriented programming principles such as inheritance, polymorphism, abstraction, and encapsulation.

2. **CO2**: Utilize inheritance, interfaces, and Java package structures to create modular and maintainable code, promoting code reusability and efficient project organization while adhering to object-oriented design principles.
3. **CO3**: Construct and handle exceptions proficiently to build fault-tolerant applications, applying exception handling mechanisms to manage errors and maintain program stability.
4. **CO4**: Develop and implement multi-threaded applications using Java's built-in thread handling mechanisms and concurrency utilities to optimize program performance and resource utilization.
5. **CO5**: Design and implement Java programs using file handling, the NIO.2 package, and Stream API to perform efficient file operations and process data streams effectively, enhancing data-driven application capabilities.
6. **CO6**: Apply the concepts of the Collection Framework and Generics to create robust and efficient programs that effectively manage and manipulate data, ensuring type safety and reducing runtime errors.

- **Self-study/Further Study components and materials:**

Unit No.	Unit/Topic Details	
1	**INTRODUCTION**	
	1.1	Overview of OOP principles: Encapsulation, Inheritance, Polymorphism, Abstraction.
	1.2	Data Types: Integer (byte, short, int, long), Floating-point (float, double), Character, Boolean.
	1.3	Variables: Dynamic Intialization, Scope and Lifetime of variable, Type Casting, Type promotion rule
	1.4	Operators: Arithmetic, Bitwise, Relational, Boolean, Assignment, ternary (three-way).
	1.5	Control Statements: if, nested ifs, Traditional Switch, Iteration statements (while, do-while, for, For-Each, Nested Loops), Jump statements (break, continue, return)
	Materials: - ➢ https://www.geeksforgeeks.org/operators-in-java/	
2	**BASICS OF ARRAY**	
	2.1	Introduction to Arrays, Types of Array
	2.2	Array Operations: Traversing arrays with for, for-each, and enhanced for loop
	2.3	Copying arrays, cloning arrays, Searching and sorting elements in an array, Jagged Arrays, Memory Representation of Arrays
	Materials: - ➢ https://www.programiz.com/java-programming/arrays ➢ https://www.geeksforgeeks.org/jagged-array-in-java/ ➢ https://www.geeksforgeeks.org/where-is-the-memory-allocated-for-arrays-in-java/	
3	**FUNDAMENTAL OF INHERITANCE**	
	3.1	Concept and need for inheritance in OOP, Types of Inheritance: Single Inheritance, Multilevel Inheritance, Hierarchical Inheritance
	Materials: - ➢ https://www.scaler.com/topics/java/inheritance-in-java/	
	FUNDAMENTAL OF POLYMORPHISM	

	3.2	Concept and need for polymorphism in OOP, Types of polymorphism, Method Overloading, Method Overriding
	Materials: - ➢ https://www.scaler.com/topics/java/polymorphism-in-java/	

Syllabus:

Unit No.	Unit/Topic Details		Hours (hr)	Evaluation Weightage (%)
1	**FUNDAMENTAL OF PROGRAMMING IN JAVA**		02	06
	1.1	History of Java.		
	1.2	Java's Magic: The Bytecode.		
	1.3	The Java Buzzwords: Simple, Object-Oriented, Robust Multithreaded, Architecture-Neutral, Interpreted and High Performance, Distributed, Dynamic.		
	1.4	Switch Expressions, Records, and Other Recently Added Features.		
2	**CLASS FUNDAMENTALS**		04	14
	2.1	Class Fundamentals: Defining classes and creating objects.		
	2.2	Instance and static variables, methods, and the new keyword, Assigning Object Reference Variables.		
	2.3	Access modifiers: public, private, protected, default (package-private).		
	2.4	Using final with classes, methods, and variables, Understanding the this keyword.		
	2.5	Constructors: Types of constructors: default and parameterized, Constructor overloading, instanceof operator, The role of the this() constructor call.		
	2.6	Nested and Inner Classes: static nested class, inner class, method-local inner class, and anonymous inner class		
	2.7	Understanding the Object Class: The Object class and its methods: equals(), hashCode(), toString(), clone(), etc. Overriding toString() and equals() in custom classes.		
	2.8	Memory Management in Java: Stack vs. heap memory, Object lifecycle and garbage collection, Manual memory management and implications.		
	2.9	Java Annotations and Meta-annotation: Overview and purpose of annotations, Custom annotation, Reflection in Java.		
3	**STRING HANDLING**		02	06
	3.1	Introduction to Strings, Methods of the String class, StringBuffer and StringBuilder, String Manipulation Techniques, Regular Expressions (Regex), String Tokenization.		
4	**INHERITANCE, INTERFACES & PACKAGES**		05	17
	4.1	Understanding super keyword, Dynamic method dispatch and runtime polymorphism, final Keyword in Inheritance.		

	4.2	Abstract Classes and Methods, Casting Objects in Inheritance.		
	4.3	Introduction to Interfaces, Interface Segregation Principle, Interface Methods and Variables, Multiple Inheritance Using Interfaces, default and static method, private interface, Functional Interfaces, Marker Interfaces.		
	4.4	Introduction to Packages, Access Modifiers and Packages, Creating and Importing Packages, Static Imports, Jar Files. Introduction to the Java Module System, Creating and Declaring a Module, Understanding Module Descriptors, Working with Multiple Modules, Services and Service Loaders.		
5	**EXCEPTIONS HANDLING**		02	06
	5.1	Introduction to Exception Handling, Types of Exceptions: Checked Exceptions, Unchecked Exceptions, Try-Catch-Finally block.		
	5.2	Throw and Throws keyword, Try-With-Resources Statement.		
	5.3	Try-With-Resources Statement, Custom Exceptions, Exception Propagation.		
6	**MULTITHREADED PROGRAMMING**		05	17
	6.1	Introduction to Multithreading, Life cycle of a thread.		
	6.2	Understanding of Thread class and the Runnable interface, Organizing threads into groups.		
	6.3	Thread Synchronization, Inter-thread Communication, Thread Priority and Scheduling, Deadlock and Thread Safety Issues.		
	6.4	Executor and ExecutorService for thread pool management, Fork/Join Framework.		
7	**FILE NIO**		04	14
	7.1	Path Interface, Files Class, File Attributes.		
	7.2	Byte and Character Stream, Serialization.		
	7.3	NIO Fundamental: Buffer, Channel, Selector. Directory Stream and File Walking.		
8	**COLLECTION FRAMEWORK AND GENERICS**		06	20
	8.1	Introduction to Collection zFramework, Collection Interface.		
	8.2	List Interface: ArrayList, LinkedList and Vector and Stack.		
	8.3	Set Interface: HashSet, LinkedHashSet and TreeSet.		
	8.4	Map Interface: HashMap, LinkedHashMap, TreeMap and Hashtable.		
	8.5	Working with Collections: Iterating through Collections, Sorting and Searching, Thread-safe Collections.		
	8.6	Introduction to Generics, Generic Classes and Methods, Bounded Type Parameters, Generics and Wildcards, Type Erasure.		
Total Hours:			**30**	

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	2	3	3	-	-	-	-	-	-	-	-	-	-	-
CO2	3	2	3	2	1	1	-	-	-	-	-	-	1	3
CO3	-	2	2	3	2	-	-	-	-	-	1	1	-	-
CO4	1	1	1	1	3	-	-	-	-	-	2	2	3	-
CO5	1	1	2	3	3	-	-	1	-	-	1	1	2	2
CO6	1	1	1	2	3	1	-	1	-	-	2	1	3	3

The correlation levels 1, 2 or 3 are as defined below:
1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High), If there is no correlation, put "-" placed.

Recommended Study Material:

❖ **Text books:**

1. "Java: The Complete Reference, Thirteenth Edition", by Herbert Schildt, McGraw Hill.
2. "OCP Oracle Certified Professional Java SE 17 Developer (Exam 1Z0-829) Programmer's Guide", by Khalid A. Mughal, Vasily A. Strelnikov, Oracle Press.

❖ **Reference Books:**

1. "Effective Java, 3rd Edition", by Joshua Bloch, Addison-Wesley Professional.
2. "Core Java Volume I--Fundamentals (Core Series)", by Cay S. Horstmann, Pearson.
3. "Head First Java, 3rd Edition", by Kathy Sierra, Bert Bates, Trisha Gee, O'Reilly Media.

Online resources:

1. https://docs.oracle.com
2. https://www.baeldung.com/
3. https://dev.java/

Care-taking Points:

- Doubts arising from self-study topics will be addressed and clarified during classroom sessions to ensure comprehensive understanding.
- Ensure that theoretical concepts are backed by hands-on practice. Every major concept, like inheritance, interface, multithreading, stream API, collections framework etc., should be complemented by coding exercises or projects to solidify understanding.
- Teach coding best practices, including clean code principles, design patterns, and proper use of Java's built-in utilities. This ensures students learn to write efficient, maintainable, and scalable code.

Practical Skill Enhancement Points:

- Practical enhancement points bridge the gap between theoretical concepts and real-world application, enabling students to gain hands-on experience in solving complex problems

- Develop the ability to implement object-oriented programming principles to design and create modular, reusable, and scalable Java applications.
- Gain hands-on experience with Java Collections, Generics, and multithreading for managing data structures and achieving concurrency in real-world scenarios.
- Enhance problem-solving skills by building programs that handle exceptions, manage file I/O, and process input/output operations efficiently.
- Acquire practical expertise in creating and debugging Java applications that integrate arrays, strings, and advanced features like NIO and interfaces.

Charotar University of Science and Technology (CHARUSAT)
Faculty of Technology and Engineering (FTE)

Subject: Engineering Graphics and Design (MEUE201) (University Elective - I)

Semester: 3rd

Teaching Scheme:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/Week	-	4	-	4	2
Marks	-	50	-	50	

Course Description:

- All engineering activities (design/ manufacturing/ operation/ servicing) for any product from any discipline involve a team of people who communicate graphically. Hence, every engineer must have exposure and some competence in presenting ideas as pictures, and be able to unambiguously interpret drawing from others. This course will help develop basic visualization competency as well as ability to representing ideas on both paper and computer.
- **URL**: https://archive.nptel.ac.in/courses/112/102/112102304/

INDUSTRY SUPPORT: All companies across all disciplines work with drawings, hence this course is relevant to all industries of all sizes.

Syllabus:

Week 1: Introduction

Week 2: Graphical Representation

Week 3: Projection Basics

Week 4: Orthographics Projections

Week 5: Auxilary and Sectional Projections

Week 6: Isometric Projections

Week 7: Working Drawings

Week 8: Introduction To CAD

Week 9: Part Modelling 1

Week 10: Part Modelling 2

Week 11: Assembly

Week 12: Design Project

Recommended Study Material:

❖ **Text books / Reference Books:**

1. SP46 Engineering Drawing Practice for Schools and Collages BIS National Drawing Code (preparation by BIS)
2. D.K. Lieu and S.A. Sorby, Visualization, Modeling, and Graphics for Engineering Design, Cengage Learning; 2nd edition
3. W.J. Luzadder and J.M. Duff, Fundamentals of Engineering Drawing, PHI learning, 11th edition

Charotar University of Science and Technology (CHARUSAT)
Faculty of Technology and Engineering (FTE)

Subject: Fundamentals Of Electrical Engineering (EEUE201) (University Elective – I)

Semester: 3rd

Teaching Scheme:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/Week	-	4	-	4	2
Marks	-	50	-	50	

Course Description:

- This course is mainly for undergraduate First-Year Engineering students from all Specializations. This course will introduce and explain the fundamental concepts of basic electrical engineering. The basic concepts of DC and AC (Single Phase and Three Phase Circuits) network analysis, first order DC transients, steady state and phasor analysis of AC networks, series and parallel resonance and magnetic coupled circuits. This course will also cover Single Phase Transformers, Three Phase Induction Machines and DC Machines. By the end of the course, the students should be able to gather high-quality knowledge of basic electrical engineering.
- **URL**: https://archive.nptel.ac.in/courses/108/105/108105112/

INDUSTRY SUPPORT: Power Grid, NTPC, NHEC, DVC and State Electricity Boards. In general, this basic course is for all engineering professionals.

Syllabus:

Week 1: Basic Concepts and Basic Laws

Week 2: Methods of Analysis

Week 3: DC Network Theorems

Week 4: Capacitors and Inductors and First Order Circuits

Week 5: Sinusoidal and Phasors

Week 6: Sinusoidal Steady-State Analysis

Week 7: AC Circuit Analysis and Network Theorems

Week 8: Series and Parallel Resonance and Magnetically Coupled Circuits.

Week 9: Three Phase Circuits and Power Measurements

Week 10: Single Phase Transformers

Week 11: Three Phase Induction Machines

Week 12: DC Machines

Recommended Study Material:

- **Text books / Reference Books:**

1. Fundamentals of Electric Circuits— C.K. Alexander and M.N.O. Sadiku (TATA McGraw-Hill)
2. Electrical Engineering: Principles and Applications- Allan R. Hambley (Pearson Education)
3. Electric Machines- Charles I. Hubert (Pearson education)
4. Electrical Machines- M.V. Deshpande (Wheeler Publishing) Engineering Circuit Analysis- W.H.Hayt and J.E. Kemmerly (McGraw-Hill)

Charotar University of Science and Technology (CHARUSAT)
Faculty of Technology and Engineering (FTE)

Subject: Environment And Development (CLUE201) (University Elective – I)

Semester: 3rd

Teaching Scheme:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/Week	-	4	-	4	2
Marks	-	50	-	50	

Course Description:

- The course will consist of theoretical application and environmental knowledge pertaining to sustainable development. The course analyses the reciprocal interaction between the physical environment, the social organization and human behaviour in the context of development. The course will introduce students with an overview of environmental ethics, debates and change and to facilitate their understanding and analysis of the inter-relationship between environment and development issues and apply them to their own experience and work. To enhance the students' knowledge of the nature of and underlying causes of the most pressing environmental concerns and to understand how these impact on the lives and livelihoods of the local community. To look at the possibilities for environmental regeneration providing an analysis of case studies of local sustainable development initiatives and community based natural resource management. After the successful completion of the course the students will be able to comprehend the complexity and various forms and dimensions of development and environment issues and ground them in current issues and real-life experiences.
- **URL**: https://nptel.ac.in/courses/109103186

Syllabus:

Week-1: Introduction: Development, economic growth and sustainable development, Basic ecosystem ecology

Week-2: Environmentalism, Environmental Movement, Environmentalism in the global south

Week-3: Approaches to environment: Ecofeminism, Feminist political ecology, Marxism and ecology

Week-4: Debates on environmental ethics: Deep ecology, Gandhi and ecology, Social ecology

Week-5: Religion, environment and conservation: Religion, environment and historical roots of ecological crisis, Biodiversity conservation ethics in Buddhism and Hinduism, Christian religion in the age of ecological crisis

Week-6: Natural resource management, Common property vs. private property,Livelihoods, forests, and conservation

Week-7: Displacement, dispossession and development: Conservation-induced displacement, Environment impact assessment and national rehabilitation & resettlement policy, Dispossession and land acquisition

Week-8: Mainstream development trajectory: Strengthening or weakening of indigenous peoples: Mining, development, and indigenous people, Competing visions of development along the Narmada, Dams, development, and resistance: case studies

Week-9: Gender and development: Development theory and gendered approach to development, Gender, environment & sustainable development

Week-10: Environment and climate change: Climate change interventions and policy framework, Eastern Himalayas and climate change

Week-11: Belief and knowledge systems, biodiversity conservation and sustainability: Ecological knowledge, biodiversity conservation and sustainability,Traditional religion and conservation of nature in Northeast India: Case study

Week-12: Local knowledge in the environment-development discourse: Indigenous knowledge, environment and development, Relevance of indigenous knowledge: case study

Recommended Study Material:

❖ **Text books / Reference Books:**

1. Arnold, David and Guha, Ramchandra, (eds.), 1997. Nature, Culture and Imperialism, New Delhi: Oxford University Press.
2. Baviskar, Amita. 1997. In the Belly of the River: Tribal Conflicts over Development in the Narmada Valley, OUP, Delhi.
3. Barnhill, David Landis & Roger S. Gottlieb. (eds.) 2001. Deep Ecology and World Religions: New Essays on Sacred Grounds. State Univ. of New York Press, Albany.
4. Bicker, Alan, Paul Sillitoe and Johan Pottier. 2004. Development and Local Knowledge: New Approaches to Issues in Natural Resources Management, Conservation and Agriculture. Routledge, London & New York.
5. Esteva, G. 1997. 'Development' in W. Sachs, ed., The Development Dictionary, Orient Longman, pp. 8-34.
6. Gadgil, Madhav and Guha, Ramchandra. 1995. Ecology and Equity: The use and Abuse of Nature in Contemporary India, New Delhi: Oxford University.
7. Gottlieb, Roger S. 2004. This Sacred Earth: Religion, Nature, Environment. Routledge, New York and London.
8. Merchant, Carolyn. 1994. Ecology: Key Concepts in Critical Theory, Humanities Press, New Jersey.
9. Ramakrishnan, P.S. 1992. Shifting Agriculture and Sustainable Development: An Interdisciplinary Study from North-Eastern India, Man and the Biosphere Series, Volume 10, UNESCO.
10. Shiva, Vandana. 1988. Staying Alive: Women, Ecology and Survival in India, Zed Press, New Delhi.

Charotar University of Science and Technology (CHARUSAT)
Faculty of Technology and Engineering (FTE)

Subject: Scientific Computing Using Matlab (ECUE201) (University Elective-I)

Semester: 3rd

Teaching Scheme:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/Week	-	4	-	4	2
Marks	-	50	-	50	

Course Description:

- In the first week of this course, an introduction to Matlab is introduced with hands on Matlab software. In the subsequent weeks, topics will be introduced followed by hands on to write the Matlab code of the same topics. This course is very useful for the students who wants to do research projects using numerical techniques to handle data or to solve differential equations.
- **URL**: https://archive.nptel.ac.in/courses/111/102/111102137/

***PREREQUISITES*:** Some knowledge of any programming language.

Syllabus:

Week 1: Introduction to Matlab

Week 2: Error estimation and methods of roots finding

Week 3: Order of convergence of various methods

Week 4: Solving System of Linear Algebraic equations

Week 5: Continuing...Solving System of Linear Algebraic equations

Week 6: Curve fitting and Interpolation

Week 7: Continuing... Curve fitting and Interpolation

Week 8: Continuing... Curve fitting and Interpolation

Week 9: Numerical differentiation

Week 10: Numerical Integration

Week 11: Numerical solution to ordinary differential equations (ODE's)

Week 12: Continuing with numerical solution to ODE's

Recommended Study Material:

❖ **Text books / Reference Books:**

1. Applied Numerical Analysis by Gerald & Wheatley, Pearson- 7th Edition, 2003.
2. Elements of Numerical Analysis by R.S. Gupta, second edition, Cambridge university press, 2015.
3. Numerical Methods Using MATLAB by Mathew & Fink, Pearson, 1998.
4. Getting started with Matlab: A quick introduction for scientist & engineers by Rudra Pratap, Oxford, 2010.

Charotar University of Science and Technology (CHARUSAT)
Faculty of Technology and Engineering (FTE)

Subject: The Joy Of Computing Using Python (CEUE201) (University Elective -I)

Semester: 3rd

Teaching Scheme:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/Week	-	4	-	4	2
Marks	-	50	-	50	

Course Description:

- A fun filled whirlwind tour of 30 hrs, covering everything you need to know to fall in love with the most sought-after skill of the 21st century. The course brings programming to your desk with anecdotes, analogies and illustrious examples. Turning abstractions to insights and engineering to art, the course focuses primarily to inspire the learner's mind to think logically and arrive at a solution programmatically. As part of the course, you will be learning how to practice and culture the art of programming with Python as a language. At the end of the course, we introduce some of the current advances in computing to motivate the enthusiastic learner to pursue further directions.
- **URL**: https://nptel.ac.in/courses/106106182

***INDUSTRY SUPPORT*:** Every software company is aware of the potential of a first course in computer science. Especially of a first course in computing, done right.

Syllabus:

Week 1: Introduction to Programming

Week 2: Introduction to Anaconda

Week 3: Lists, Loops, Crowd Computing

Week 4: Practice is the key

Week 5: Dictionaries, Sorting and Searching

Week 6: Substitution Cipher -The science of secrecy

Week 7: Snakes and Ladders - Not on the Board

Week 8: Tuples- Python Data Structure

Week 9: Natural Language Processing - Author Stylometry

Week 10: Flames, Date compression

Week 11: Browser Automation Watsapp using Python

Week 12: Page Rank - How does Google Work ?

Charotar University of Science and Technology (CHARUSAT)

Faculty of Technology and Engineering (FTE)

Subject: Introduction To Quantum Computing (ITUE201) (University Elective – I)

Semester: 3rd

Teaching Scheme:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/Week	-	4	-	4	2
Marks	-	50	-	50	

Course Description:

- Quantum computing is fast emerging as one the key disruptive technologies of our times. It is a fundamentally new computing paradigm that has the potential to efficiently solve certain challenging problems which cannot be solved efficiently in a classical setting. IBM has made significant investment in this technology and is recognized as a leader in this space. This course will provide introduction to Quantum Computation, starting with basic concepts such as superposition and entanglement, to discussing the quantum circuit model of computation and basic Quantum algorithms that demonstrate the power of computing with quantum bits. We will also introduce the idea of quantum error correction to mitigate the effects of noise in today's quantum devices. We will have full hands-on sessions for each concept taught using Qiskit, a pythonic way of programming and the IBM Circuit Composer.
- **URL**: https://nptel.ac.in/courses/106106232

***PRE-REQUISITES*:** Basic Linear algebra is a pre-requisite. (You can brush-up from any crash course, ex: https://www.khanacademy.org/math/linear-algebra).

***INDUSTRY SUPPORT*:** IBM, Microsoft, Intel, TCS, Infosys, CQC, Honeywell, Google, Riggetti, D-Wave and several others would be interested and value this course as IBM has been the pioneer in Quantum technologies both in hardware and software and have several Quantum computers available as well as the entire software stack is available.

Syllabus:

Week 1: Introduction and IBM Quantum Perspective, Q Mission in India – Invited talk, Quantum Computing Applications, Quantum Computing Basics

Week 2: IBM Quantum Composer and Quantum Lab using Qiskit

Week 3: Quantum Algorithms-I (Oracles, Deustch Jozsa), Quantum Algorithms-II (Grover's Algorithm with Hands-on)

Week 4: Quantum Error Correction, NISQ era Quantum Algorithms (VQE/QAOA and industrial applications)

Recommended Study Material:

- **Text books / Reference Books:**

1. Qiskit Textbook: https://qiskit.org/textbook/preface.html
2. YouTube Quantum learning series: https://www.youtube.com/playlist?list=PLOFEBzvs-Vvp2xg9-POLJhQwtVktlYGbY
3. Quantum Computation and Quantum Information, Textbook by M. A. Nielsen and I. Chuang, Cambridge University Press (2010).

Charotar University of Science and Technology (CHARUSAT)
Faculty of Technology and Engineering (FTE)

Subject: Python For Data Science (CSUE201) (University Elective – I)

Semester: 3rd

Teaching Scheme:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/Week	-	4	-	4	2
Marks	-	50	-	50	

Course Description:

- The course aims at equipping participants to be able to use python programming for solving data science problems.
- **URL**: https://nptel.ac.in/courses/106106212

PRE-REQUISITES: Knowledge of basic data science algorithms.

Syllabus:

Week 1: BASICS OF PYTHON SPYDER (TOOL): Introduction Spyder, Setting working Directory, Creating and saving a script file, File execution, clearing console, removing variables from environment, clearing environment, Commenting script files, Variable creation, Arithmetic and logical operators, Data types and associated operations.

Week 2: Sequence data types and associated operations: Strings, Lists, Arrays, Tuples, Dictionary, Sets, Range, NumPy: ndArray.

Week 3: Pandas dataframe and dataframe related operations on Toyota Corolla dataset: Reading files, Exploratory data analysis, Data preparation and preprocessing, Data visualization on Toyoto Corolla dataset using matplotlib and seaborn libraries: Scatter plot, Line plot, Bar plot, Histogram, Box plot, Pair plot, Control structures using Toyota Corolla dataset: if-else family, for loop, for loop with if break, while loop, Functions.

Week 4: Case Study: Regression- Predicting price of pre-owned cars, Classification- Classifying personal income.

Recommended Study Material:

- **Text books / Reference Books:**

1. Introduction to linear algebra - by Gilbert Strang.
2. Applied statistics and probability for engineers – by Douglas Montgomery.
3. Mastering python for data science, Samir Madhavan.

Charotar University of Science and Technology (CHARUSAT)
Faculty of Technology and Engineering (FTE)

Subject: An Introduction To Artificial Intelligence (AIUE201) (University Elective - I)

Semester: 3rd

Teaching Scheme:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/Week	-	4	-	4	2
Marks	-	50	-	50	

Course Description:

- The course introduces the variety of concepts in the field of artificial intelligence. It discusses the philosophy of AI, and how to model a new problem as an AI problem. It describes a variety of models such as search, logic, Bayes nets, and MDPs, which can be used to model a new problem. It also teaches many first algorithms to solve each formulation. The course prepares a student to take a variety of focused, advanced courses in various subfields of AI.
- **URL**: https://nptel.ac.in/courses/106102220

PRE-REQUISITES: Data Structures, Probability

INDUSTRY SUPPORT: Software companies

Syllabus:

Week 1 : Introduction: Philosophy of AI, Definitions

Week 2 : Modeling a Problem as Search Problem, Uninformed Search

Week 3 : Heuristic Search, Domain Relaxations

Week 4 : Local Search, Genetic Algorithms

Week 5 : Adversarial Search

Week 6 : Constraint Satisfaction

Week 7 : Propositional Logic & Satisfiability

Week 8 : Uncertainty in AI, Bayesian Networks

Week 9 : Bayesian Networks Learning & Inference, Decision Theory

Week 10 : Markov Decision Processes

Week 11 : Reinforcement Learning

Week 12 : Introduction to Deep Learning & Deep RL

Recommended Study Material:

- **Text books / Reference Books:**

1. Stuart Russell & Peter Norvig, Artificial Intelligence: A Modern Approach, Prentice-Hall, Third Edition (2009) (required).
2. Ian GoodFellow, Yoshua Bengio & Aaron Courville, Deep Learning, MIT Press (2016).

Charotar University of Science and Technology (CHARUSAT)
Faculty of Technology and Engineering (FTE)

Subject: Money And Banking (BMUD201) (University Elective-I)

Semester: 3rd

Teaching Scheme:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/Week	-	4	-	4	2
Marks	-	50	-	50	

Course Description:

- This is an introductory course on Indian money and banking. It begins with a discussion on the functions of money, types of money and interest rates. Module 2 discusses the money supply process in India with definitions of Indian monetary aggregates, money multipliers and sources of high-powered money in India. Module 3 extensively describes Indian money market instruments. Then, Module 4 follows with a description of the various functions of the Reserve Bank of India including its traditional and non-traditional monetary policy instruments, and Liquidity Adjustment mechanisms. Finally, Module 5 concludes with an overview of the Indian banking system, with particular reference to the commercial and co-operative banks.
- **URL**: https://nptel.ac.in/courses/109104076

PRE-REQUISITES: Intermediate Macroeconomics

Syllabus:

Week 1: What is Money - Functions of Money, Kinds of Money, Interest Rates and Return

Week 2: Money Supply - Traditional Monetary Aggregates, New Monetary Aggregates, Velocity of Money, Monetary Base, Credit Multiplier, Money Multiplier

Week 3: Money Market - Indian Financial Institutions, Call Money, Commercial Paper, Certificate of Deposit, Treasury, Bill, Repo, Commercial Bill

Week 4: Reserve Bank of India - Traditional and Non-traditional Functions of the RBI, Liquidity Adjustment Facility, Primary Dealers, DFHI

Week 5: Banking Institutions - Indian Banking System, Lead Bank Scheme, Action Plan and Service Area Approach, Regional, Rural Banks, Assets and Liabilities of Scheduled, Commercial Banks, Co-operative Banks

Recommended Study Material:

- **Text books / Reference Books:**

1. L M Bhole and Jitendra Mahakud, Financial Institutions and Markets, TataMcGraw-Hill, 2009.

2. F S Mishkin, The Economics of Money, Banking, and Financial Markets, Prentice Hall, 2007.
3. S B Gupta, Monetary Economics, S Chand Limited, 1988.
4. Economic Survey, Ministry of Finance, Government of India.
5. RBI Bulletin, www.rbi.org

Charotar University of Science and Technology (CHARUSAT)
Faculty of Technology and Engineering (FTE)

Subject: Human Computer Interactions (CAUD203) (University Elective – I)

Semester: 3rd

Teaching Scheme:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/Week	-	4	-	4	2
Marks	-	50	-	50	

Course Description:

- Why are things so hard to use these days? Why does this thing I just bought work? Why is this web site so hard to use? Why is the phone app so confusing? Why are users not liking my design? Why is my app not getting popular? These are frustrations that we have all faced from systems not designed with people in mind. The question this course will focus on is: how can we design human-centered systems that people find useful and usable? This course is an introduction to designing, prototyping, and evaluating user interfaces. If you can take only one course in Human-Computer Interaction, this is the course for you.
- **URL**: https://archive.nptel.ac.in/courses/106/106/106106177/

PRE-REQUISITES: Interest in interfaces, curiosity in fixing interface issues.

INDUSTRY SUPPORT: Any company which is interested in HCI will be interested in recruiting the students finishing the course.

Syllabus:

Week 1: Components of HCI Types of interfaces Design process

Week 2: Contextual inquiry Importance of users / talking to users Task analysis

Week 3: Sketching Low & hi fidelity prototyping

Week 4: Mental models

Week 5: Usability evaluation Think aloud, observing users Modelling users, expert valuations

Week 6: Information visualization

Week 7: HCI & mobility New faces of HCI

Week 8: Refresher for all modules seen in the course

Charotar University of Science and Technology (CHARUSAT)
Faculty of Technology and Engineering (FTE)

Subject: Basics Of Health Promotion and Education Intervention (PTUD191)
(University Elective - I)

Semester: 3rd

Teaching Scheme:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/Week	-	4	-	4	2
Marks	-	50	-	50	

Course Description:

- This course is designed to provide the learners a comprehensive understanding about Health Promotion and Education Interventions. Health Promotion and Education Interventions are key competencies for public health professionals. After completion of this course the participants will be able to define and understand the needs for health promotion and education interventions. The participants will also be able to plan, implement, and evaluate Health Promotion and Education Interventions.
- **URL**: https://onlinecourses.nptel.ac.in/noc22_ge18/preview

PRE-REQUISITES: Graduate degree in health sciences.

INDUSTRY SUPPORT: NGOs working in health and development sector, public health organizations, Health services sector of state and central governments (It will be beneficial for the Medical Officers also, who are involved in Public Health system).

Syllabus:

Week 1: Concepts of health promotion including history

Week 2: Health behavior, health communication and Health Literacy

Week 3: Information Education Communication (IEC), Behavior Change Communication (BCC), and Social and Behavior Change Communication (SBCC), and their applications in different settings (including role of social determinants of health)

Week 4: Need assessment for health promotion (including health behavior models)

Week 5: Planning and implementing a HPE intervention

Week 6: Designing of messages and pretesting

Week 7: Materials and methods

Week 8: Evaluation of HPE intervention

Recommended Study Material:

- **Text books / Reference Books:**

1. "Principles and Foundations of Health Promotion and Education" by James McKenzie.
2. "Health Behavior: Theory, Research, and Practice" by Karen Glanz.
3. "Health Behavior Change" by Prestwich, Kenworthy, and Conner.

Charotar University of Science and Technology (CHARUSAT)
Faculty of Technology and Engineering (FTE)

Subject: First Aid Masterclass- A complete guide to first aid (NRMD251)
(University Elective - I)

Semester: 3rd

Teaching Scheme:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/Week	-	4	-	4	2
Marks	-	50	-	50	

Course Description:

- This comprehensive course equips learners with essential first aid skills and practical emergency response techniques applicable in a wide range of real-life scenarios. Designed to build confidence and competence, the course focuses on quick, effective interventions that can be life-saving in critical situations. With a balance of theoretical understanding and practical skills, this 30-hour course prepares students to act swiftly and responsibly in various emergency situations, empowering them to save lives both in personal and community settings.
- **URL**: https://www.udemy.com/course/first-aid-masterclass/?couponCode=ST12MT90625AI

Syllabus:

Unit No.	Title of Unit	Prescribed Hours
1.	**Introduction to first aid:** • Treat choking using the Heimlich Maneuver • Treat bleeding wounds using different bandages and tourniquets. • Using tourniquets from random cloth rags Correct order of action – Setting priorities in emergency	05
2.	**Hot and cold related emergency:** • Dehydration • Hyperthermia • Frostbite or hypothermia, including how to avoid them.	05
3.	**CPR:** • CPR (using the latest guidelines), with 2 different methods. • AED (Automated External Defibrillator) to make our CPRs even more effective.	10

4.	**Injuries:** • Head injuries • Burns, • Fractures • Chest injuries • Animal bites	05
5.	**Medical emergencies:** • Heart attack, stroke, allergy attack, epilepsy • Carry an injured person, use different equipment or build your very own first aid kit	05
	Total:	**30**

Recommended Study Material:

❖ **Text books / Reference Books:**

1. Lewis Heitkemper, “Medical-Surgical Nursing”, 7th edition, Mosby, Elsevier-2009.
2. Black J.M. Hawk, J.H. (2005), “Medical Surgical Nursing Clinical Management for Positive Outcomes”, 7th edition, Elsevier.
3. "First Aid Manual" by Indian Red Cross Society.
4. "Textbook of First Aid and Emergency Nursing" by PK Lakshmi.
5. "Manual of First Aid: Management of Injuries and Emergencies" by S.K. Joshi.

B. Tech. (CE/CSE/IT) Programme

SYLLABI

(Semester – 4)

CHAROTAR UNIVERSITY OF SCIENCE AND TECHNOLOGY

Charotar University of Science and Technology (CHARUSAT)
Faculty of Technology and Engineering (FTE)

Subject: Fundamentals of Database Management Systems (ITUC202)

Semester: 4th

Teaching Scheme:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/Week	3	4	-	7	5
Marks	100	50	-	150	

Course Description:

The Database Management Systems (DBMS) course introduces fundamental concepts and techniques for designing, implementing, and managing databases. It covers relational and NoSQL databases, ER modeling, schema refinement, and query languages like SQL and MongoDB. Advanced topics such as transaction management, concurrency control, database security, and query optimization are also explored. The course integrates practical applications with foundational theory and introduces basics of data analytics to prepare students for modern database challenges and real-world problem-solving.

Course Objectives:

1. Provide a comprehensive understanding of the relational model and relational algebra.
2. Equip students with knowledge of database design theory, including normalization and related algorithms, to ensure efficient organization of database tables.
3. Familiarize students with real-world database systems, their architectures, and components, including examples like Oracle, MySQL, MS Access, and SQL Server, MongoDB.
4. Develop proficiency in Database Management Systems, the standard language for interacting with and managing databases.

Course Outcomes:

By the end of the course, students will be able to:

1. **CO1**: Understanding of the design of database solutions to data management problems.
2. **CO2**: Effectively use the Entity Relationship Diagram for the representation of conceptual schemas.
3. **CO3**: Identify functional dependencies and apply normalization.
4. **CO4**: Understand the transaction concept and its role in transaction processing systems.
5. **CO5**: Demonstrate data retrieval procedures using the Data Manipulation Language.
6. **CO6**: Index databases and query optimization.

- **Self-study/Further Study components and materials:**

Unit No.	Unit/Topic Details	
1	**INTRODUCTORY CONCEPTS OF DATABASE MANAGEMENT SYSTEM**	
	1.1	Introduction to DBMS and its applications
	1.2	Traditional database Vs. File System
	Materials: ➢ https://www.geeksforgeeks.org/introduction-of-dbms-database-management-system-set-1/	

Syllabus:

Unit No.	Unit/Topic Details		Hours (hr)	Evaluation Weightage (%)
1	**INTRODUCTORY CONCEPTS OF DATABASE MANAGEMENT SYSTEM**		05	11
	1.1	Purpose of databases, data independence		
	1.2	Database architecture: levels, mappings, DBA roles		
	1.3	Structure of relational databases: schema, instances, integrity constraints		
	1.4	Introduction to NoSQL databases: types (document, key-value, columnar, graph), comparison with RDBMS		
	1.5	Entity-Relationship Model (Entities, attributes, relationships, and cardinality)		
	1.6	Weak entities, aggregation, generalization, specialization		
	1.7	Converting ER diagrams to relational schemas		
	1.8	Schema-less design principles in NoSQL databases		
2	**FORMAL RELATIONAL QUERY LANGUAGES**		05	11
	2.1	Relational Algebra: Selection, projection, set operations, joins		
	2.2	Tuple Relational Calculus and Domain Relational Calculus		
	2.3	NoSQL queries: find, aggregate, and pipelines in MongoDB		
3	**SCHEMA REFINEMENT AND NORMALIZATION**		12	18
	3.1	Functional dependencies, attribute closures		
	3.2	Normalization: 1NF, 2NF, 3NF, BCNF, 4NF, 5NF		
	3.3	Decomposition using Functional Dependencies		
	3.4	Schema-less design in NoSQL databases: trade-offs and use cases		
4	**TRANSACTION, CONCURRENCY AND RECOVERY MANAGEMENT**		12	18
	4.1	Transaction properties (ACID)		
	4.2	Serializability: conflict serializability and testing		
	4.3	CAP theorem and BASE properties in NoSQL		
	4.4	Locking protocols, deadlock handling, timestamp ordering		
	4.5	Log-based recovery, shadow paging, and checkpoints		
	4.6	Replication and sharding in distributed NoSQL databases (e.g., MongoDB, DynamoDB)		

5	**DATABASE SECURITY AND QUERY OPTIMIZATION**		10	15
	5.1	SQL security: roles, privileges, authorization, and views		
	5.2	Security in NoSQL: encryption, AWS DynamoDB IAM policies		
	5.3	Query cost evaluation, join strategies, query transformation		
	5.4	Aggregation pipelines and query optimization		
6	**INDEXING AND HASHING**		10	15
	6.1	Ordered indices: B+ Trees, bitmap indexing		
	6.2	Hashing: static and dynamic hashing		
	6.3	Indexing in NoSQL databases		
7	**OVERVIEW OF DATA ANALYTICS**		06	12
	7.1	Data analytics and its importance		
	7.2	Data types, data pre-processing, and exploratory data analysis		
	7.3	Integrating databases with data analytics tools (e.g., Python libraries: pandas, NumPy)		
	7.4	Data aggregation and preparation		
		Total Hours:	**60**	

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3	3	2	3	-	-	-	2	3	-	-	-	-
CO2	3	3	3	2	3	-	-	-	2	3	-	-	-	-
CO3	3	3	3	3	3	-	-	-	2	3	-	-	-	-
CO4	3	3	3	3	3	-	-	-	2	3	-	-	-	-
CO5	3	2	3	2	3	-	-	-	2	3	2	3	-	-
CO6	3	3	3	3	3	-	-	-	2	3	-	-	-	-

The correlation levels 1, 2 or 3 are as defined below:
1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High), If there is no correlation, put "-" placed.

Practical Objectives:

- Implement SQL queries using MySQL for relational databases.
- Perform NoSQL operations using MongoDB or DynamoDB.
- Design ER diagrams and convert them to relational schemas.
- Normalize a given database schema to higher normal forms.
- Query optimization and cost evaluation.
- Explore basics of data analytics with sample datasets.

Recommended Study Material:

❖ **Text book:**

1. "Database System Concepts", by Silberschatz, Korth, and Sudarshan, 6th Edition, Publisher: McGraw Hill, ISBN: 978-0078022159.
2. "Fundamentals of Database Systems", by Elmasri and Navathe, 7th Edition, Publisher: Pearson, ISBN: 978-0133970777.

❖ **Reference book:**

1. “NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence”, by Pramod J. Sadalage and Martin Fowler.
2. “SQL: The Complete Reference”, by James R. Groff and Paul N. Weinberg.
3. “Database Management Systems”, by Raghu Ramakrishnan and Johannes Gehrke, 3rd Edition.
4. “MongoDB: The Definitive Guide”, by Shannon Bradshaw, Eoin Brazil, and Kristina Chodorow, 3rd Edition.

❖ **Additional Online Resources:**

1. MongoDB Official Documentation: https://www.mongodb.com/docs/
2. AWS DynamoDB Documentation: https://docs.aws.amazon.com/dynamodb
3. W3Schools SQL Tutorial: https://www.w3schools.com/sql/
4. Database Normalization Explained (Khan Academy): https://www.khanacademy.org/computing

❖ **Online Courses**

1. Data Base Management System, by Prof. P. Das and Prof. S. Chattopadhyay, IIT Kharagpuro: https://onlinecourses.nptel.ac.in/noc22_cs91/preview
2. Databases and SQL for Data Science (Coursera - IBM): https://www.coursera.org/learn/sql-data-science
3. MongoDB Basics (MongoDB University): https://university.mongodb.com/
4. SQL for Data Analysis (Udacity): https://www.udacity.com/course/sql-for-data-analysis--ud198

Charotar University of Science and Technology (CHARUSAT)
Faculty of Technology and Engineering (FTE)

Subject: Fundamentals of Software Engineering (CEUC201)

Semester: 4th

Teaching Scheme:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/Week	4	0	-	4	4
Marks	100	0	-	100	

Course Pre-requisites:

- **Programming Basics:** Knowledge of at least one programming language (e.g., Java or Python).
- **Data Structures and Algorithms:** Understanding of basic data structures (arrays, lists, trees) and algorithms (sorting, searching).

Course Description:

This course introduces the principles, methodologies, and tools of software engineering. It covers the entire software development lifecycle, from requirement analysis to maintenance, emphasizing the importance of risk management, project planning, quality assurance, and emerging trends. The curriculum integrates theoretical concepts with hands-on learning through lectures, tutorials, case studies, and individual exercises, fostering a strong understanding of agile development, design principles, and testing strategies.

Course Objectives:

1. To develop a comprehensive understanding of software engineering principles, methodologies, and tools.
2. To familiarize students with the software development lifecycle, including requirement analysis, design, implementation, testing, and maintenance.
3. To teach students risk management techniques and the importance of quality assurance in software projects.
4. To enable students to apply agile methodologies and project management practices for efficient software development.
5. To explore emerging trends in software engineering and equip students with skills for contemporary software development practices.

Course Outcomes:

By the end of the course, students will be able to:

1. **CO1**: Understand basics about software engineering principles, methods and practices and to analyze software requirement specification Prepare, SRS (Software Requirement Specification) document and SPMP (Software Project Management Plan) document.

2. **CO2**: Able to understand modern Agile Development and Design thinking Concept of Industry.
3. **CO3**: Apply the concept of Functional Oriented and Object Oriented Approach for Software Design, to explain the software design strategies and to apply software measurement and metrics using Function point, Cyclomatic complexity.
4. **CO4**: Formulate problem by following Software Testing Life Cycle. Apply various testing techniques and test plan in. Design Manual Test cases for Software Project. Use automation testing tool students will be able test the software.
5. **CO5**: Recognize how to ensure the quality of software product, different quality standards and software review techniques.
6. **CO6**: Analyze software risk with estimation parameters such as cost, effort, and schedule/duration and understand the concepts of software maintenance, reengineering, reverse engineering, software configuration management.

- **Self-study/Further Study components and materials:**

Sr. No.	Components Details
1	**Independent Case Analysis: -** Students will analyse case studies outside of class, focusing on real-world applications of software engineering principles.
	Evaluation Strategy: - Submission of written reports on selected case studies.
	Skills Developed: - Critical thinking, analytical skills, and the ability to connect theory to practice.

Syllabus:

Unit No.	Unit/Topic Details		Hours (hr)	Evaluation Weightage (%)
1	**INTRODUCTION TO SOFTWARE AND SOFTWARE ENGINEERING**		08	15
	1.1	Importance of Software Engineering in Technology, The Evolving Role of Software		
	1.2	Software: A Crisis on the Horizon and Software Myths		
	1.3	Case Study: A failed software project		
	1.4	Discussion: Lessons Learned, Risk Management in Software Engineering (Importance of risk management in software projects, Types of Risks: Technical, Project, Organizational, and External risks, Risk Identification: Techniques for identifying risks in the software development lifecycle, Risk Assessment, Risk Mitigation Strategies, Monitoring and Controlling Risks, Risk Management Techniques in Process Models)		
	Self-study Components: Process Models in Software Engineering			
2	**AGILE DEVELOPMENT**		06	10
	2.1	Agility and Agile Development		
	2.2	SCRUM, Kanban, Lean		
	2.3	Case Study: Suitability of Agile		

	Self-study Component: XP-Programming			
3	**MANAGING SOFTWARE PROJECT**		08	10
	3.1	Software Metrics: Process, Product, and Project		
	3.2	Software Project Estimations (Agile Estimation, Function Point Analysis (FP), COCOMO II, Monte Carlo Simulation for Estimation)		
	3.3	Software Project Planning, Scheduling, and Tracking		
	Self-study Component: Simulation Exercise: Managing a project timeline			
4	**REQUIREMENT ANALYSIS AND SPECIFICATION**		07	10
	4.1	Understanding the Requirement		
	4.2	Requirement Gathering Techniques (Interview, Brainstorming, Questionnaires, Survey, Observation, Record Review)		
	4.3	Requirement Specification (SRS), Requirement Analysis and Requirement Elicitation, User Story		
	4.4	Use case, Diagram, Use case Narratives		
	Self-study Component: Role-Play: Eliciting requirements			
5	**SOFTWARE DESIGN**		08	10
	5.1	Design Concepts and Design Principles		
	5.2	Function Oriented Design (DFD, ERD)		
	5.3	Object Oriented Design (Introduction to OOD, Unified Modeling Language (UML))		
	5.4	UI/UX Design Thinking, Cohesion and Coupling in Software Design, Concept of Design Document		
	Self-study Component: Case Study: Evaluating design decisions			
6	**SOFTWARE TESTING**		08	15
	6.1	Coding Standard and coding Guidelines, Code Review		
	6.2	Testing Strategies: Black Box and White Box, Testing Techniques (Unit, Integration, System)		
	6.3	Test Case & Test Suites Design		
	Self-study Component: Automated Testing Tool: Selenium			
7	**SOFTWARE QUALITY ASSURANCE**		06	10
	7.1	Quality Concepts and Software Quality Assurance		
	7.2	Software Reviews (Formal Technical Reviews)		
	7.3	Software Reliability, Overview of Quality Standards		
8	**EMERGING TRENDS IN SOFTWARE ENGINEERING**		04	10
	8.1	AI driven Software Development, Software 2.0		
9	**SOFTWARE MAINTENANCE AND CONFIGURATION MANAGEMENT**		05	10
	9.1	Introduction to Software Maintenance Types of Maintenance (Corrective, Adaptive, Preventive, Perfective)		
	9.2	Software Configuration Management (SCM), Version Control in Software Maintenance		
	9.3	Change Management in Software Maintenance, Build and Release Management		
	9.4	Automation in Software Maintenance and Configuration Management, Documentation and Knowledge Management in Maintenance, Monitoring and Reporting in Software		

		Maintenance, Re-Engineering, Reverse Engineering, Forward Engineering		
	Self-study Component: Maintenance challenges in legacy systems, Version Control and Change Control tools			
		Total Hours:	**60**	

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3	-	2	1	-	-	3	-	-	-	-	2	-
CO2	3	2	2	1	2	1	-	-	2	-	1	-	2	-
CO3	3	2	3	2	1	1	1	-	2	1	1	1	-	-
CO4	3	3	2	-	1	-	-	-	1	-	-	1	2	-
CO5	3	-	-	-	-	1	-	2	1	1	1	1	1	1
CO6	-	2	-	1	-	-	1	-	2	2	1	2	1	-

The correlation levels 1, 2 or 3 are as defined below:
1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High), If there is no correlation, put "-" placed.

Recommended Study Material:

❖ **Text books:**

1. Roger S. Pressman, "Software engineering- A practitioner's Approach", McGraw-Hill International Editions.
2. Rajib Mall, "Fundamentals of software Engineering", Prentice Hall of India.

❖ **Reference Books:**

1. Ian Sommerville, "Software engineering", Pearson education Asia.
2. Pankaj Jalote, "An Integrated Approach to Software Engineering" by Springer.
3. John M Nicolas," Project Management for Business, Engineering and Technology", Elsevier.

❖ **Web material:**

1. http://vlabs.iitkgp.ac.in/se/

❖ **Tools List:**

- Estimation Tools
    1) Costar/System Star Tool
- Software Project Management Plan (SPMP)
    1) Microsoft Project 2003/2007
    2) OpenProject
    3) LibrePlan
    4) ProjectLibre
- Design Tools

1) Microsoft Visio 2007/2010/2013/2016
2) Pencil Tool
3) ForeUI
4) UMLet 14.2
5) SmartDraw
6) OpenSource Tool

- Testing Tools
    1) WinRunner
    2) Silk Runner
    3) Load Runner
    4) Selenium/Appium
- Different Case Tools and Testing Tools
    1) QTP
    2) QTest
    3) IBM Rational Functional tester
    4) MSC (message sequence chart)
    5) SDL (specification and description language)
    6) TTCN (testing and test control notation)
    7) TTCN-3

Care to be Taken Points:

- Ensure all group activities promote collaboration and equitable participation.
- Emphasize real-world applications to keep students engaged.
- Use diverse case studies to cover various domains and challenges.
- Students need to be wok in group of 4/5 and need to work on real-life case study which will be shared by faculty.
- There are series of activity in class / labs for practical learning, especially for Biding of Projects, Request for Proposal, Design Thinking, Agile Methodology, Project Management, Demonstration of Tools, etc.

Charotar University of Science and Technology (CHARUSAT)
Faculty of Technology and Engineering (FTE)

Subject: Computer Organization and Architecture (CEUC202)

Semester: 4th

Teaching Scheme:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/Week	3	0	-	3	3
Marks	100	0	-	100	

Course Pre-requisites:

- Digital Electronics

Course Description:

This course offers a fundamental understanding of the concepts in Computer Organization and architecture. It provides students with an understanding of the design of fundamental blocks used for building a computer system and interfacing techniques of these blocks to achieve different configurations of an "entire computer system". The course emphasizes on computer design & parameter measurement, computer arithmetic used in commuting, cache memory configuration and optimization principles.

Course Objectives:

1. To provide introduction to Instruction Set Architecture and Practical exposure through simulation tools/Microprocessor Kits
2. To explore the basic concepts of computer organization & computer architecture design, Computer System Components: Processor, Memory, and Performance evaluation
3. To provide insight details in Processor Components: Control Unit, Registers, Caches Memory, ALU, and Instruction Execution Unit.

Course Outcomes:

By the end of the course, students will be able to:

1. **CO1**: Understand and Design of Arithmetic circuit, Logic circuit, Shift circuit and Control Unit circuit by using elements of digital logic.
2. **CO2**: Visualize and understand the working of CPU, different instruction formats, addressing modes, pipeline and vector processing.
3. **CO3**: Use various metrics to calculate and analyze clock periods, performance, and instruction throughput of single-cycle, multi-cycle, and pipelined implementations of a simple instruction set.
4. **CO4**: Understand and apply r's and r-1's complement in Addition, Subtraction and Multiplication of signed and unsigned numbers.
5. **CO5**: Show how cache design parameters affect the performance of program and Map a virtual address into a physical address.

Syllabus:

Unit No.	Unit/Topic Details		Hours (hr)	Evaluation Weightage (%)
1	**REGISTER TRANSFER AND MICROOPERATIONS BASIC COMPUTER ORGANIZATION AND DESIGN**		13	29
	1.1	Register Transfer Language		
	1.2	Register Transfer		
	1.3	Bus and Memory Transfers		
	1.4	Arithmetic, Logic and Shift Micro-Operations		
	1.5	Arithmetic Logic Shift Unit		
	1.6	Instruction Codes		
	1.7	Computer Registers, Computer Instructions		
	1.8	Timing and Control, Instruction Cycle		
	1.9	Memory Reference, Instructions, Input-Output and Interrupt		
	1.10	Complete Computer Description, Design of Basic Computer, Design of Accumulator Logic		
2	**CENTRAL PROCESSING UNIT & PIPELINE AND VECTOR PROCESSING**		12	26
	2.1	General Register Organization & Stack Organization,		
	2.2	Instruction Formats, Addressing Modes		
	2.3	Parallel Processing & Pipelining		
	2.4	Arithmetic Pipeline & Instruction Pipeline		
	2.5	RISC Pipeline		
3	**PERFORMANCE MEASURES**		04	09
	3.1	Performance and Cost, Purchasing perspective		
	3.2	Design perspective Notions of Performance: Latency and throughput, Performance and time, computer clocks, Computing CPU time and cycles, Improving Performance		
	3.3	Linking instruction, cycles and time, CIPS and MIPS examples, Computer Benchmarks, Sources of Benchmark: SPEC 89 and SPEC 95		
	3.4	Amdahl's law, Estimating performance improvements, poor performance metrics		
4	**COMPUTER ARITHMETIC**		08	18
	4.1	Introduction: Binary, Octal, Decimal, Hexadecimal representation		
	4.2	Integer Numbers: Sign-Magnitude,1's complement,2's complement		
	4.3	Addition, Subtraction & Multiplication Algorithm		
5	**MEMORY ORGANIZATION**		08	18
	5.1	Memory construction, size, speed, cost and data unit, Tradeoffs between them		
	5.2	PROM, EEPROM, DRAM, SRAM, Memory Technologies, Hierarchical organization		
	5.3	Principle of locality, Simple Cache organization, Miss rate, block size, cache policies		

	5.4	Cache Organization: Mapping alternatives- direct, associative and set associative, processor performance with cache, memory organization and miss penalty		
	5.5	Policies for read, load, fetch, replacement and write, How Caches work, Size of tags, Performance analysis examples		
			Total Hours:	**45**

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3	2	3	2	1	-	-	-	-	-	-	-	-
CO2	3	3	3	3	2	1	-	-	-	-	-	-	-	-
CO3	3	3	3	3	3	1	-	-	-	-	-	-	-	-
CO4	3	2	2	3	2	1	-	-	-	-	-	-	-	-
CO5	3	3	3	3	3	1	-	-	-	-	-	-	-	-

The correlation levels 1, 2 or 3 are as defined below:
1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High), If there is no correlation, put "-" placed.

Recommended Study Material:

- **Text books:**

1. "Computer System Architecture", by Morris Mano, 3rd Edition, Prentice Hall.
2. John L. Hennesy & David A. Patterson, "Computer Organization and Design MIPS Edition: The Hardware/Software Interface", (The Morgan Kaufmann Series in Computer Architecture and Design).

- **Reference Books:**

1. "Structured Computer Organization", A. S. Tananbum, Pearson Education.
2. "Computer Organization & Architecture-Designing for Performance", William Stalling, Pearson Prentice Hall, 8th Edition.
3. "The Essentials of Computer Organization and Architecture", by Linda Null, Julia Lobur.
4. "Computer Architecture & Organization", by John P Hayes, McGraw-Hill.
5. "Computer Architecture: Pipelined and Parallel Processor Design", by Michael J. Flynn, 4th Edition.

Online courses:

- High Performance Computer Architecture, Prof. Ajit Pal, IIT Kharagpur, https://nptel.ac.in/courses/106105033/ (For cache memory and Pipelining).

Web Material:

1. http://pages.cs.wisc.edu/~markhill/cs354/Fall2008/notes/flpt.apprec.html
2. https://www.youtube.com/watch?v=qlH4-oHnBb8
3. https://www.ebookbou.edu.bd/Books/Text/SST/DCSA/dcsa_2301/Unit-08.pdf
4. https://www.youtube.com/playlist?list=PLxCzCOWd7aiHMonh3G6QNKq53C6oNXGr X

Charotar University of Science and Technology (CHARUSAT)
Faculty of Technology and Engineering (FTE)

Subject: Design and Analysis of Algorithms (ITUE204)

Semester: 4th

Teaching Scheme:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/Week	4	2	-	6	5
Marks	100	50	-	150	

Course Pre-requisites:

- Programming Language Knowledge (C / C++ / Java / Python)
- Data Structures and Algorithms

Course Description:

This course provides a detailed understanding of algorithm design, analysis, and optimization techniques. It covers algorithm design paradigm such as Divide & Conquer, Greedy approach, Dynamic programming, Backtracking, Branch & Bound. The advanced algorithms concepts such as approximation algorithms, randomized algorithm and NP-completeness are included in the course. The course emphasizes real-world applications, preparing students to address computational challenges in domains like machine learning, cryptography, bioinformatics, and optimization.

Course Objectives:

1. Introduce the fundamental concepts of algorithms, including time and space complexity, to analyze algorithm performance using asymptotic notations.
2. Understand the different algorithm design techniques using its characteristics and performance.
3. Enhance problem solving skills for solving computation problem using efficient algorithm design technique.
4. Understand complexity classes, their significance and challenges for solving real-world problem.

Course Outcomes:

By the end of the course, students will be able to:

1. **CO1**: Analyze the time and space complexity of algorithms to evaluate efficiency of algorithm for solving computational problems.
2. **CO2**: Apply efficient algorithm design technique such as Divide & Conquer, Greedy, Dynamic Programming, Backtracking, Branch & Bound to solve computational problem.
3. **CO3**: Apply Graph algorithms to find shortest path, exploring connectivity and

4. **CO4**: Identify whether a given problem is tractable or intractable, and apply reduction theory to relate the complexity of different problems.

- **Pre-requisites online video / course:**
  - Introduction to Algorithms and Analysis, by Dr. S. Mukhopadhyay, IIT Kharagpur, https://archive.nptel.ac.in/courses/106/105/106105164/.
  - Introduction to Algorithms, MIT OCW Algorithms Course, by Prof. Demaine, https://ocw.mit.edu/courses/6-006-introduction-to-algorithms-spring-2020/.
  - Design and Analysis of Algorithms, by Prof. A. Ranade, IIT Bombay, https://archive.nptel.ac.in/courses/106/101/106101060/.
  - Design and Analysis of Algorithms, by Prof. M. Mukund, Chennai Mathematical Institute, https://nptel.ac.in/courses/106106131.
  - Computational Complexity Theory, Prof. R. Tewari, IIT Kanpur, https://archive.nptel.ac.in/courses/106/104/106104227/.

- **Self-Study/Further Study components and materials:**

Unit No.	Unit/Topic Details	
1	**INTRODUCTION (10 HOURS)**	
	1.1	Study the history of algorithms and their evolution to understand their significance in computing.
	1.2	Practice analyzing algorithms using asymptotic notations in real-world systems, like estimating algorithm performance for large datasets.
	Materials: ➢ Types of Asymptotic Notation – GeeksforGeeks • https://www.geeksforgeeks.org/types-of-asymptotic-notations-in-complexity-analysis-of-algorithms/	
2	**DIVIDE AND CONQUER (08 HOURS)**	
	2.1	Explore Karatsuba algorithm for fast multiplication using divide and conquer approach.
	2.2	Explore Maximum Subarray Sum problem using divide and conquer approach.
	Web Resources: ➢ Divide and Conquer- GeeksforGeeks • https://www.geeksforgeeks.org/divide-and-conquer/ ➢ Divide and Conquer Problem • https://www.geeksforgeeks.org/karatsuba-algorithm-for-fast-multiplication-using-divide-and-conquer-algorithm/ ➢ Divide and Conquer Problem • https://www.geeksforgeeks.org/maximum-subarray-sum-using-divide-and-conquer-algorithm/	
3	**GREEDY ALGORITHMS (10 HOURS)**	
	3.1	Analyze the trade-offs between greedy algorithms and dynamic programming for solving optimization problems.
	3.2	Explore machine learning applications, such as feature selection using greedy methods.
	Web Resources: ➢ Huffman Coding - GeeksforGeeks • https://www.geeksforgeeks.org/huffman-coding-greedy-algo-3/	
4	**DYNAMIC PROGRAMMING (12 HOURS)**	

	4.1	Explore advanced dynamic programming problems like the traveling salesman problem (TSP) and implement solutions.
	4.2	Investigate how dynamic programming is applied in reinforcement learning and real-time optimization problems.
	Web Resources: ➢ Edit Distance • https://www.geeksforgeeks.org/edit-distance-dp-5/	
5	**BACKTRACKING, BRANCH & BOUND (05 HOURS)**	
	5.1	Explore permutation of string, Hamiltonian cycle problems using backtracking approach.
	5.2	Explore Travelling Salesman Problem (TSP) using Branch and Bound approach.
	Web Resources: ➢ Backtracking – GeeksforGeeks • https://www.geeksforgeeks.org/backtracking-algorithms/ ➢ Branch and Bound – GeeksforGeeks • https://www.geeksforgeeks.org/branch-and-bound-algorithm/	
6	**STRING MATCHING ALGORITHMS (05 HOURS)**	
	6.1	Investigate the role of string matching algorithms in search engines and bioinformatics.
	6.2	Study advanced techniques for approximate string matching used in genomics and text analysis.
	Web Resources: ➢ String Matching Algorithms – GeeksforGeeks • https://www.geeksforgeeks.org/pattern-searching/ ➢ KMP Algorithm - W10L5_String Matching: Knuth-Morris-Pratt Algorithm • https://www.youtube.com/watch?v=IrVjHDGjmKk	
7	**NP-COMPLETENESS, APPROXIMATION AND RANDOMIZED ALGORITHMS (10 HOURS)**	
	7.1	Explore current research trends in approximating solutions for NP-complete problems.
	7.2	Learn heuristic approaches for practical applications like circuit design and AI planning.
	7.3	Learn about the use of randomized algorithms in cryptography and blockchain technologies.
	7.4	Study advanced approximation techniques, such as local search and simulated annealing.
	Web Resources: ➢ P vs NP Problem – GeeksforGeeks • https://www.geeksforgeeks.org/introduction-to-np-completeness/	

Syllabus:

Unit No.	Unit/Topic Details		Hours (hr)	Evaluation Weightage (%)
1	**INTRODUCTION**		10	20
	1.1	Introduction to Algorithms, characteristics of Algorithm, Model for Analysis of Algorithm: Random Access Machine (RAM)		
	1.2	Analysis of Algorithm: Best, Average and Worst Case Analysis Algorithm Analysis Techniques: Mathematics, Empirical and Asymptotic Analysis.		

<table>
<tr><td></td><td>1.3</td><td>Asymptotic Notations: Big-O, Big-Theta and Big-Omega</td><td></td><td></td></tr>
<tr><td></td><td>1.4</td><td>Recurrence Relations: Linear Recurrence Relations, Divide and Conquer Recurrences, Substitution Recurrences, Homogeneous Recurrences, Non-Homogeneous Recurrences</td><td></td><td></td></tr>
<tr><td></td><td colspan="2">Prerequisites:
Understanding of basic mathematical concepts such as induction and summation.
Basic knowledge of programming constructs like loops and conditional statements.
Web Resources:
Types of Asymptotic Notations – GeeksforGeeks
https://www.geeksforgeeks.org/types-of-asymptotic-notations-in-complexity-analysis-of-algorithms/</td><td></td><td></td></tr>
<tr><td rowspan="4">2</td><td colspan="2">DIVIDE AND CONQUER</td><td rowspan="4">8</td><td rowspan="4">15</td></tr>
<tr><td>2.1</td><td>General structure for Divide and Conquer Approach</td></tr>
<tr><td>2.2</td><td>Design and Analyze the Problems using Divide and Conquer Approach:
Binary Search
Merge Sort
Quick Sort
Strassen's Matrix Multiplication</td></tr>
<tr><td colspan="2">Prerequisites:
Familiarity with basic sorting algorithms like quicksort and merge sort.
Familiar with Recursive approach and Recurrence Relation.
Web Resources:
Binary Search - GeeksforGeeks
https://www.geeksforgeeks.org/binary-search/
Quick Sort– GeeksforGeeks
https://www.geeksforgeeks.org/quick-sort-algorithm/
https://www.geeksforgeeks.org/time-and-space-complexity-analysis-of-quick-sort/</td></tr>
<tr><td rowspan="4">3</td><td colspan="2">GREEDY ALGORITHMS</td><td rowspan="4">10</td><td rowspan="4">15</td></tr>
<tr><td>3.1</td><td>General Characteristics of Greedy Algorithms, Greedy choice property</td></tr>
<tr><td>3.2</td><td>Design and Analyze the Problems using Greedy Algorithm:
Making Change Problem
Knapsack Problem
Job Scheduling Problem
Huffman Coding
Optimal Merge Pattern</td></tr>
<tr><td>3.3</td><td>Minimum Spanning Tree (MST) algorithms:
Analysis of Minimum Spanning Tree Algorithms (Kruskal’s Algorithm, Prim’s Algorithm)
Single Source Shortest Path Algorithm (Dijkstra Algorithm)</td></tr>
<tr><td colspan="2">Prerequisites:
Familiarity with basic sorting algorithms.</td></tr>
</table>

<table>
<tr><td></td><td colspan="2">• Understanding of basic decision-making strategies and graph representations.
Web Resources:
➢ MST Algorithms
• https://www.nptelvideos.com/video.php?id=990
• https://www.geeksforgeeks.org/what-is-minimum-spanning-tree-mst/</td><td></td><td></td></tr>
<tr><td rowspan="4">4</td><td colspan="2">DYNAMIC PROGRAMMING</td><td rowspan="4">12</td><td rowspan="4">25</td></tr>
<tr><td>4.1</td><td>The Principle of Optimality, Key Characteristics of Dynamic Programming: Memoization and Tabulation.</td></tr>
<tr><td>4.2</td><td>Design and Analyze the Problems using Dynamic Programming:
• Binomial Coefficient
• Making Change Problem
• Knapsack Problem
• Matrix Chain Multiplication
• Longest Common Subsequence
• Subset Sum Problem</td></tr>
<tr><td colspan="2">Prerequisites:
• Knowledge of recursion and memorization techniques.
• Basic problem-solving using arrays and matrices.
Web Resources:
➢ Dynamic Programming
• https://www.geeksforgeeks.org/dynamic-programming/
➢ Knapsack Problem
• https://www.geeksforgeeks.org/0-1-knapsack-problem-dp-10/</td></tr>
<tr><td rowspan="4">5</td><td colspan="2">BACKTRACKING, BRANCH & BOUND</td><td rowspan="4">05</td><td rowspan="4">10</td></tr>
<tr><td>5.1</td><td>General characteristics of Backtracking, Design and Analyze the Problems using Backtracking Approach: N- Queen Problem, Knapsack Problem, M-coloring problem</td></tr>
<tr><td>5.2</td><td>Introduction to Branch and Bound, Design and Analyze the Problems using Branch & Bound Approach: Assignment Problem</td></tr>
<tr><td colspan="2">Prerequisites:
• Knowledge of tree construction.
• Basic problem-solving using arrays and matrices.
Web Resources:
➢ Assignment Problem using Branch and Bound – GeeksforGeeks
• https://www.geeksforgeeks.org/job-assignment-problem-using-branch-and-bound/</td></tr>
<tr><td rowspan="3">6</td><td colspan="2">STRING MATCHING ALGORITHMS</td><td rowspan="3">05</td><td rowspan="3">10</td></tr>
<tr><td>6.1</td><td>String Matching Algorithms:
Naive approach, Knuth-Morris-Pratt (KMP), Boyer-Moore</td></tr>
<tr><td colspan="2">Prerequisites:
• Basic knowledge of string operations (concatenation, substring).
• Familiarity with automata theory and regular expressions.
Web Resources:
➢ KMP Algorithm
• https://www.geeksforgeeks.org/kmp-algorithm-for-pattern-searching/</td></tr>
</table>

7	**NP-COMPLETENESS, APPROXIMATION AND RANDOMIZED ALGORITHMS**		10	05
	7.1	Classes P, NP, NP-hard		
	7.2	Polynomial-time reductions		
	7.3	Examples: SAT, 3-SAT, vertex cover, Hamiltonian cycle		
	7.4	Introduction to Approximation Algorithms, Travelling salesman Problem		
	7.5	Introduction to Randomized Algorithms, Randomized Quick sort		
	Prerequisites: • Familiarity with computational complexity concepts. • Basic understanding of graph theory and optimization problems. **Web Resources:** ➢ Approximation Algorithms • https://www.geeksforgeeks.org/approximation-algorithms/ • https://onlinecourses.nptel.ac.in/noc24_cs97/preview ➢ Randomized Algorithms • https://www.geeksforgeeks.org/randomized-algorithms/ ➢ Randomized QuickSort • https://www.geeksforgeeks.org/quicksort-using-random-pivoting/ • https://www.youtube.com/watch?v=hOIZpznZWMw			
			Total Hours:	**60**

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	3	2	-	-	-	-	-	-	-	3	-	-
CO2	3	3	3	3	-	-	-	-	-	-	-	2	-	-
CO3	3	3	3	3	-	-	-	-	-	-	-	2	-	-
CO4	3	2	3	3	-	-	-	-	-	-	-	2	-	-

The correlation levels 1, 2 or 3 are as defined below:
1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High), If there is no correlation, put "-" placed.

Recommended Study Material:

❖ **Text books:**

1. Thomas H. Coreman, Charles E. Leiserson, Ronald Rivest and Clifford Stein, "Introduction to Algorithms", MIT Press.
2. A Gills Brassard, Paul Brately, "Fundamental of Algorithms", Prentice Hall of India.

❖ **Reference Books:**

1. Jon Kleninberg, Eva Tardos, "Algorithm Design", Cornell University, Pearson Addison Wesley.
2. Jeff Erickson, "Algorithms", University of Illinois Urbana-Champaign.
3. Ellis Horowitz, S. Sahni and S. Rajasekarn, "Fundamental of Computer Algorithms", Computer Science Press.
4. https://jeffe.cs.illinois.edu/teaching/algorithms/book/Algorithms-JeffE.pdf

Charotar University of Science and Technology (CHARUSAT)
Faculty of Technology and Engineering (FTE)

Subject: Software Group Project (ITUP201 / CEUP201 / CSUP201)

Semester: 4th

Teaching Scheme:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/Week	0	2	-	2	2
Marks	0	50	-	50	

Course Pre-requisites:

- Basic knowledge of programming, data structures, DBMS, software engineering, object-oriented concepts, etc. is expected.

Course Description:

This course focuses on collaborative software development through a group project, where students work in teams to analyze, design, and implement a system. The project may be product-based, process-based, or research-based, and it involves a comprehensive feasibility study, design documentation, development, and testing plan, including performance and security considerations. Students will engage in continuous evaluation through weekly progress reviews, presentations, and written reports as per NEP-2020 Policy. The course aims to develop students' technical skills, teamwork, and project management abilities while preparing them to address real-world problems using industry-standard tools and methodologies.

Course Objectives:

1. To develop the ability to work collaboratively in a group and apply theoretical knowledge to the solution of the real-world problems.
2. To conduct a comprehensive analysis, design, development, and testing of software or systems, considering performance and security aspects.
3. To enhance project management and time management skills through structured project milestones and deadlines.
4. To foster effective technical skill through written reports, soft-skill through presentations, demonstrations and problem-solving skill.
5. To encourage innovation and critical thinking in problem-solving within interdisciplinary project environments.

Course Outcomes:

By the end of the course, students will be able to:

1. **CO1**: Identify a solution, critically evaluate and justify proposed design solution.
2. **CO2**: Manage learning & self-development including development of organizational skills, time management, effective use of scientific literature and discriminating use of Web resources.

3. **CO3**: Apply a wide range of principles and tools available to the software developer such as choice of the algorithm, language, software libraries etc.
4. **CO4**: Write and test programs using appropriate test cases.
5. **CO5**: Solve communication issues in large, complex software projects and Structure & communicate ideas effectively orally. Also Prepare & deliver coherent and structured verbal and written technical reports.
6. **CO6**: Evaluate system in terms of general quality attributes and possible trade-offs presented within the given problem/system

Course Outline:

- Students will form groups of 3 to 4. If the group size is more than four students, approval from the HOD and Principal must be taken in the beginning.
- Interdisciplinary group formation across branches or institutes is allowed based on the nature of the project.
- Students must submit their project definition for approval through a formal departmental process.
- Guide and Co-guide, if required, allocation will be completed by the department within a week of completion of project definition submission. Students may have freedom to select a guide, but not as a matter of right in the institute.
- After approval, students are required to report their progress weekly to their assigned guide.
- The project work must include feasibility study, requirement analysis, design, implementation, testing, and consideration of performance and security aspects depending on the nature of the project.
- Continuous evaluation will take place during laboratory/project hours based on common rubric-based assessment criteria defined by the department / Institute.
- Domain expertise should be considered as one of the key criteria for allocating project guides. The department may define additional criteria as needed, however the process remains transparent.
- External evaluation will include a presentation, demonstration, and viva lasting at least 30 minutes per group.

Instructional Methodology and Pedagogy:

- Students are encouraged to select innovative, interdisciplinary, product-based projects, research-oriented, or process-driven projects beyond conventional management systems.
- Students are expected to acquire relevant domain knowledge aligned with the project's objectives, beyond just technical implementation.
- Each group will be allocated a guide based on domain expertise; external guides from other CHARUSAT institutes may be involved if needed.
- Weekly meetings with the guide are mandatory for progress tracking, feedback, and continuous assessment.
- Project development should follow the complete Software Development Life Cycle (SDLC), including all five categories or relevant methodology depending on the project type.
- Tools like Git and GitHub should be used to manage and track project progress; guides may provide demonstrations of these tools.
- A structured report/research paper must be prepared and submitted at the end of the semester in the prescribed format.

- Students are required to submit a plagiarism report (similarity index < 40%) signed by the guide at least 15 days before the external examination.
- Students must carry their internal review card during internal and external reviews, duly filled and signed as per the guidelines.

Role of a Guide (Mentor):

1. Initial Guidance
   - Help students in selecting a suitable and innovative project/research topic.
   - Ensure that the title is clear, specific, relevant, and feasible.
   - Confirm the project matches the course/program objectives.
2. Regular Interaction & Monitoring
   - Meet students regularly to track their progress.
   - Ensure students maintain discipline, attend meetings, and follow timelines.
   - Provide suggestions and corrections at each phase of the project.
3. Project Planning & Review Support
   - Assist students in project planning (e.g., Gantt charts, milestones).
   - Check if the work is progressing as per the planned stages.
   - Guide during all project/research reviews (Review 1, Review 2, Final Review).
4. Technical Support
   - Help students choose proper tools, technologies, and frameworks.
   - Check the implementation of core features and ensure best coding practices.
   - Suggest testing methods and evaluate test results.
5. Documentation & Report Checking
   - Review the project documentation, including research papers, reports, and user manuals.
   - Ensure students properly document installation steps, usage, testing, and references.
   - Check presentation slides and research writing format.
6. Encourage Teamwork & Ethics
   - Encourage collaboration, communication, and equal contribution among team members.
   - Promote ethical practices like avoiding plagiarism, maintaining privacy, and giving credit.
7. Support for External Communication (if applicable)
   - Guide students in client interactions or stakeholder feedback (if involved).
   - Suggest improvements based on feedback from others.
8. Final Evaluation Support
   - Help students prepare for final demonstrations.
   - Evaluate their learning, challenges faced, and how they solved them.
   - Encourage reflection and suggest future scope for their work.

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3	1	2	-	2	2	2	2	-	1	3	3	2
CO2	3	3	1	2	1	2	2	2	2	-	1	3	3	2
CO3	3	1	3	3	3	1	2	2	3	-	2	3	3	2
CO4	3	1	1	3	1	-	-	1	2	-	2	2	2	2
CO5	3	-	-	-	-	-	2	3	3	3	3	2	2	1
CO6	3	2	1	2	1	-	-	1	2	-	1	1	3	1

The correlation levels 1, 2 or 3 are as defined below:
1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High), If there is no correlation, put "-" placed.

Charotar University of Science and Technology (CHARUSAT)
Faculty of Technology and Engineering (FTE)

Subject: Foundation of Data Science and Analytics (CSUE203) (Elective-I)

Semester: 4th

Teaching Scheme:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/Week	4	2	-	6	5
Marks	100	50	-	150	

Course Pre-requisites:

- Programming language and Database Management System

Course Description:

This course offers a fundamental understanding of the concepts, tools, and techniques in Data Science and Analytics. Students will explore various stages of data collection, pre-processing, exploratory data analysis (EDA), visualization, and analytics. Through theoretical discussions and practical applications, they will learn to handle different types of data, employ data pre-processing methods, and apply visualization techniques for insightful decision-making including the predictive analysis. The course also emphasizes ethical issues and real-world problem-solving across various domains.

Course Objectives:

1. To develop students' proficiency in data collection techniques and pre-processing methods for real-world applications.
2. To demonstrate different statistical methods for Exploratory Data Analysis (EDA).
3. To develop problem solutions using data analytics techniques with appropriate visualizations and interpreting results through effective storytelling with data.

Course Outcomes:

By the end of the course, students will be able to:

1. **CO1**: Apply pre-processing techniques on data and prepare it for analysis.
2. **CO2**: Conduct exploratory data analysis using statistical methods and feature engineering techniques.
3. **CO3**: Apply appropriate data visualization methods and create dashboards that communicate insights clearly.
4. **CO4**: Solve real-world problems by applying analytics techniques, including descriptive, diagnostic, predictive, and prescriptive models.

- **Pre-requisites online video / course:**
  - Introduction to R: https://www.udemy.com/course/introduction-to-r/

- **Self-study components and materials:**

Unit No.	Unit/Topic Details	
1	**INTRODUCTION**	
	1.1	Introduction to Data Science and Data Analytics
	1.2	Applications and Emerging Trends of Data Science
	1.3	Ethical and legal issues in data science
	1.4	Data collection techniques – Primary and Secondary Data Collection and types of data
	1.5	Data Cleaning - Missing Values, Noisy Data, Data Cleaning as a Process
	Materials: ➢ https://www.ibm.com/topics/data-science ➢ https://www.youtube.com/watch?v=RBSUwFGa6Fk ➢ https://www.microsoft.com/en-us/ai/responsible-ai ➢ https://www.coursera.org/articles/data-collection-methods ➢ https://www.splunk.com/en_us/blog/learn/data-structured-vs-unstructured-vs-semi-structured.html ➢ "Data Mining Concepts and Techniques", by Jiawei Han, Micheline Kamber, Jian Pei, Third Edition (pp. 88 to 91)	
2	**EXPLORATORY DATA ANALYTICS (EDA)**	
	2.1	Descriptive statistics – Mean, Median, Mode, Range, Standard Deviation
	Materials: ➢ https://www.youtube.com/watch?v=SplCk-t1BeA	
3	**DATA VISUALIZATION**	
	3.1	Importance of Data Visualization
	3.2	Types of Visualizations
	Materials: ➢ https://www.coursera.org/articles/data-visualization ➢ How to turn data into stories by Cole Nussbaumer Knaflic | Talks at Google, https://www.youtube.com/watch?v=Hfx1X9WSGYQ ➢ "The Truthful Art: Data, Charts, and Maps for Communication", by Alberto Cairo (Read pp. 41-65 & 121- 149)	
4	**DATA ANALYTICS**	
	4.1	Importance of Data Analytics
	Materials: ➢ https://online.hbs.edu/blog/post/types-of-data-analysis	

Syllabus:

Unit No.	Unit/Topic Details		Hours (hr)	Evaluation Weightage (%)
1	**OVERVIEW AND DATA PRE-PROCESSING**		12	20
	1.1	Difference between Data analysis and analytics		
	1.2	Data Science Lifecycle		
	1.3	Roles and Challenges in Data Science		
	1.4	Types of data attributes		
	1.5	Major tasks in Data Pre-processing		

	1.6	Data Integration - Entity Identification Problem, Tuple Duplication		
	1.7	Data Transformation – Min-max normalization, z-score normalization, and Decimal scaling; Binning		
	1.8	Data encoding – Label encoding and one-hot encoding		
	1.9	**Pre-processing** Semi-structured data - Parsing and Extraction, Converting semi-structured data into structured formats Unstructured data - Text: Text Cleaning, Stopword Removal and Tokenization; Image: Image Resizing, Cropping, Histogram Equalization, Noise Reduction, and Normalization		
2	**EXPLORATORY DATA ANALYTICS (EDA)**		18	30
	2.1	Hypothesis Testing		
	2.2	Data Sampling - Simple Random Sampling, Systematic Sampling, Stratified Sampling, Stratified Random Sampling, Cluster sampling		
	2.3	Univariate, bivariate, and overview of multivariate analysis		
	2.4	Introduction to ANOVA, Two-way ANOVA		
	2.5	Feature selection - Filter Methods: Correlation and chi-square, Wrapper Methods: Recursive Feature Elimination, Forward/Backward Selection Embedded Methods		
	2.6	Feature Extraction - Principal Components Analysis		
	2.7	Regression analysis – Simple and Multiple		
	2.8	Outliers and Anomalies		
	2.9	Text data: Bag of Words, Word cloud Image data: Image Augmentation, Edge Detection and Contour Analysis		
3	**DATA VISUALIZATION**		12	20
	3.1	Principles of Effective Visualization		
	3.2	Chart Selection Strategies		
	3.3	Multivariate Data Visualization - Scatter Plot Matrix, Heatmap, Bubble Plot, Parallel Coordinate Plots		
	3.4	High-Dimensional Data Visualization - t-SNE		
	3.5	Geospatial Data Visualization - Choropleth Maps, Dot Density Maps, Contour Maps, 3D Maps, Topographic map, Cartogram map, Hexagonal binning map		
	3.6	Time Series Visualization - Line Chart, Candlestick Chart		
	3.7	Interactive Visualizations: Interactive Features and Charts, Real-Time Data Integration, Dashboard tool - Google looker studio & Tableau		
	3.8	Storytelling with Data		
4	**DATA ANALYTICS**		18	25
	4.1	Types of Data Analytics		
	4.2	Descriptive Analytics & Diagnostic Analytics - Root Cause Analysis, Drill-Down Analysis		
	4.3	Predictive Analytics: Predictive Modeling – Classification using Logistic regression and k-nearest neighbors, clustering using k-means; Time-Series Analysis – Moving average, ARIMA model		

	4.4	Prescriptive Analytics - Optimization Techniques – Types of optimization models, Linear programming problem, Non-linear programming problem - Gradient descent; Simulation Modeling - Fundamentals of simulation, simulation types, Monte-Carlo simulation		
			Total Hours:	**60**

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3	2	3	2	-	-	-	-	-	-	2	-	-
CO2	3	3	3	3	2	-	-	-	-	-	-	2	-	-
CO3	2	3	3	3	3	-	-	-	-	3	-	3	-	-
CO4	3	3	3	3	3	-	-	-	-	-	-	3	-	-

The correlation levels 1, 2 or 3 are as defined below:
1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High), If there is no correlation, put "-" placed.

Recommended Study Material:

- **Text books:**

1. "Data Mining Concepts and Techniques", by Jiawei Han, Micheline Kamber, Jian Pei, Third Edition, Elsevier, 2012. ISBN 978-0-12-381479-1
2. "R in Action: Data Analysis and Graphics with R", by Robert I. Kabacoff, Manning, 2nd Edition

- **Reference Books:**

1. "Beginning Data Science in R: Data Analysis, Visualization, and Modelling for the Data Scientist", by Thomas Mailund, Apress publishing
2. Wu, J., and Coggeshall, S. Foundations of Predictive Analytics. CRC Press.
3. "Prescriptive Analytics - The Final Frontier for Evidence-Based Management and Optimal Decision Making", by Dursun Delen, Pearson

- **Online resources:**

1. Advanced R Programming for Data Analytics in Business, by Prof. A. Tripathi, IIT Kanpur, https://onlinecourses.nptel.ac.in/noc24_mg113/preview
2. Practitioners Course In Descriptive, Predictive And Prescriptive Analytics , by Prof. D. Philip and Prof. A. Oberoi, IIT Kanpur https://onlinecourses.nptel.ac.in/noc19_mg42/preview
3. Coursera, Introduction to R Programming for Data Science, https://www.coursera.org/learn/introducton-r-programming-data-science

- **Tools:**

  - RStudio, Orange, Google looker studio, Power BI, and Tableau

Care-taking Points:

- **Clarification of Self-Study Topics:** Doubts arising from self-study topics will be addressed and clarified during classroom sessions to ensure comprehensive understanding.
- **Case Study-Based Pedagogy:** Case study-based learning will be the primary teaching method to provide practical exposure and hands-on problem-solving experience in data science and analytics.
- **Hands-On Learning:** Practical exercises, real-world case studies, and projects will form an essential part of the course to help students apply theoretical knowledge to practical situations.
- **Industry-Relevant Tools and Techniques:** Students will work with industry-standard tools and technologies to develop the skills needed to solve real-world data science problems.
- **Regular Assessments:** Continuous assessment through quizzes, assignments, case study and project work will be conducted to track progress and reinforce learning.
- **Expert/Guest Lectures:** Interaction with industry experts through guest lectures, webinars, or workshops will offer insights into the latest trends and applications in data science.

Guidelines for the Practical Problems:

The problems should be designed to ensure that students gain:

- **Hands-on Experience with Real-World Datasets:** Students will work with diverse datasets from different non-technical domains, applying theoretical knowledge to real-world problems.
- **Focus on Data Cleaning and Pre-processing:** Students will handle challenges like missing data, outliers, and imbalanced datasets to ensure the data is clean and ready for analysis.
- **Visualization of Data Insights:** Students will develop effective visualizations to represent complex data and derive actionable insights.
- **Solution Development:** Problem solution should be developed using R programming language.

Examples:

- Analyze air quality data to find pollution trends, identify key contributing factors, and explore potential correlations with environmental and seasonal variables.
- Develop model for galaxy classification into spiral and elliptical categories using the Sloan Digital Sky Survey (SDSS) data. [https://machinelearningbook.com/wp-content/uploads/2015/07/FMLPDA_SampleChapter_CaseStudy_GalaxyClassification.pdf]
- Analyze transactional data to identify patterns in product sales and customer purchasing behaviour.
- Identify unusual patterns in credit card transactions.

Charotar University of Science and Technology (CHARUSAT)
Faculty of Technology and Engineering (FTE)

Subject: Foundation of Embedded System (CSUE204) (Elective-I)

Semester: 4th

Teaching Scheme:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/Week	4	2	-	6	5
Marks	100	50	-	150	

Course Pre-requisites:

- Digital Electronics, Basic Electronics, C Programming language

Course Description:

This course will introduce the students about fundamental knowledge of an Embedded System and its Case studies-based examples. Students will learn about core components of Embedded system like sensors, actuators, processors and controllers. The course also emphasizes on Embedded Firmware Development and Embedded Operating system. Overview of Wearable Devices will help to understand the application of Embedded Systems.

Course Objectives:

This course emphasizes on comprehensive treatment of embedded hardware and embedded operating systems along with case studies, in tune with the requirements of Industry. The objective of this course is to enable the students to understand embedded-system programming and apply that knowledge to design and develop embedded solutions.

Course Outcomes:

By the end of the course, students will be able to:

1. **CO1**: To understand the fundamental knowledge and development aspect of embedded system.
2. **CO2**: Able to usage of Embedded Protocols used in embedded system design and interface peripheral devices
3. **CO3**: To understand different steps involved in the design and development of embedded firmware
4. **CO4**: To understand fundamental concepts of embedded operating systems.
5. **CO5**: To understand the concept of wearable devices employed for real-life applications.

- **Self-study components and materials:**

Unit No.	Unit/Topic Details	
1	**OVERVIEW OF EMBEDDED SYSTEMS**	
	1.1	History of Embedded System

	1.2	Application areas of Embedded System – Digital Camera, Washing Machine, Smart watch
	Materials: ➢ https://www.totalphase.com/blog/2023/03/a-history-of-embedded-systems-the-apollo-guidance-computer-and-beyond/?srsltid=AfmBOor5eP1Y4kRwI-6wZLjRcDhkuOtbIlkD1JDZ9QAXdY_UJ-Io_rWZ ➢ https://www.theengineeringprojects.com/2016/11/examples-of-embedded-systems.html ➢ https://www.essaycompany.com/essays/information-technology/embedded-systems-and-home-applications-information-technology-essay ➢ https://www.futurelearn.com/info/courses/embedded-systems/0/steps/64773	
2	**EMBEDDED FIRMWARE DESIGN AND DEVLOPMENT**	
	2.1	Overview of Software and Firmware terminology
	2.2	Trade-off between C and Embedded C
	2.3	Native Compilers and Cross Compilers
	Materials: ➢ https://www.totalphase.com/blog/2023/01/embedded-software-vs-embedded-firmware/?srsltid=AfmBOoq0mD1B_VR7QEYqHXF0L5GhK4oMbN6EBlmwXZYvbjD_lHrHGhi ➢ https://www.geeksforgeeks.org/what-are-the-differences-between-c-and-embedded-c/ ➢ https://www.geeksforgeeks.org/difference-between-native-compiler-and-cross-compiler/ ➢ https://labs.dese.iisc.ac.in/embeddedlab/hardware-and-software-setup/#:~:text=What%20is%20Toolchain%3F,produced%20by%20the%20GNU%20Project	

Syllabus:

Unit No.	Unit/Topic Details		Hours (hr)	Evaluation Weightage (%)
1	**OVERVIEW OF EMBEDDED SYSTEMS**		05	10
	1.1	Introduction		
	1.2	Embedded System and General-Purpose System		
	1.3	Classification		
	1.4	Characteristics		
	1.5	Embedded Systems design metrics & challenges		
	1.6	Case Study of Embedded System Design		
2	**THE CORE EMBEDDED SYSTEM**		15	25
	2.1	Elements of an Embedded System		
	2.2	General Purpose and Domain Specific Processors		
	2.3	Overview of Microprocessor and Microcontroller		
	2.4	RISC and CISC Philosophy		
	2.5	Harvard and Von Neumann Architecture		
	2.6	Concept of Big-Endian and Little Endian		
	2.7	Load – Store Architecture and Instruction Pipelining		
	2.8	Memory Selection for Embedded System		
	2.9	Overview of IC Technology		

	2.10	Sensors and Actuators: Sensors – Temperature & Humidity, Soil Moisture, ECG, IR, Motion, Obstacle. Actuators – LED, LCD, DC Motor, Stepper Motor, Servo Motor		
3	**EMBEDDED PROTOCOLS**		15	25
	3.1	Overview of Embedded Communication		
	3.2	On-board Communication Interface: I2C, SPI, UART and 1 – wire Interface		
	3.3	External Communication Interface: RS-232 and RS-485 Standard, USB, Bluetooth, Wi-Fi, ZigBee		
4	**EMBEDDED FIRMWARE DESIGN AND DEVLOPMENT**		15	20
	4.1	Overview of Embedded Firmware		
	4.2	Embedded Firmware Components: Source File, Object File, Linker, Loader, Locator, Debugger		
	4.3	Concept of Integrated Development Environment (IDE)		
	4.4	Programming in Embedded C: I/O Functions, Pointer to Array, Structure Padding and Bit Fields, Macro Substitutions, Bit Manipulation Operations, Re-entrant and Recursive Functions, Dynamic Memory Allocation		
	4.5	Embedded C Standardization using MISRA C		
	4.6	Embedded C Program Documentation as per Industry Standard		
5	**EMBEDDED OPERATING SYSTEMS**		05	15
	5.1	Overview of Operating System		
	5.2	Types of Operating System		
	5.3	Classification of Operating System		
	5.4	Concept of Task, Process and Thread		
	5.5	Concept of Multithreading and Multitasking		
6	**OVERVIEW OF WEARABLE DEVICES**		05	05
	6.1	Role of Wearable Devices		
	6.2	Attributes of Wearable Devices		
	6.3	Challenges and Opportunities for Designing Wearable Devices		
	6.4	Software Design flow of Wearable Devices using Unified Modeling Language (UML), Use Case Diagram, Sequence Diagram, Collaboration Diagram, State Chart Diagram, Activity Diagram		
		Total Hours:	**60**	

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3	2	2	1	-	-	-	-	-	-	2	-	-
CO2	3	3	3	3	2	-	-	-	-	-	-	3	-	-
CO3	3	3	3	3	2	-	-	-	-	-	-	2	-	-
CO4	3	2	3	3	1	-	-	-	-	-	-	2	-	-
CO5	2	3	2	2	3	-	-	-	-	-	-	3	-	-

The correlation levels 1, 2 or 3 are as defined below:
1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High), If there is no correlation, put "-" placed.

Recommended Study Material:

❖ **Text books:**

1. "Introduction to Embedded Systems", by Shibu K.V. Tata –MGHill, Publication.
2. "Embedded Systems", by Raj Kamal, Fourth Edition, McGraw Hill Publication, 2020. ISBN 978-9353168025.

❖ **Reference Books:**

1. David E. Simon, "An Embedded Software Primer", Addition-Wesely.
2. Frank vahid& Tony D.Givargis, "System Design: A unified Hardware/Software Introduction", Addison Wesley 2002.
3. Wayne Wolf, "Computers as components Principles of embedded computing".
4. Tammy Noergaard, "Embedded Systems Architecture", Elsevier.
5. Daniel W.Lewis, "Fundamentals of Embedded Software: Where C and Assembly Meet", Prentice Hall.
6. Subhas C. Mukhopadhyay, "Wearable Electronics Sensors-For Safe and Healthy Living", Springer International Publishing, 2015.

Online resources:

1. https://takeuforward.org/
2. https://www.tutorialspoint.com/embedded_systems/index.htm
3. Introduction to Embedded System Design, by Prof. D. Gadre and Prof. B. Subudhi, IIT Jammu - https://onlinecourses.nptel.ac.in/noc20_ee98/preview

Practical Skill Enhancement Points:

- **Code Optimization:** writing efficient and optimized code to minimize resource usage, maximize performance, ensure system stability and simplify testing and maintenance.
- **Hardware Interaction:** knowing how to use programming languages to directly interact with and control hardware components effectively.
- **Platform-Specific Libraries**: Leverage platform-specific libraries and frameworks to simplify development and improve code maintainability.
- **Debugging Techniques:** Master debugging techniques in different programming languages to identify and resolve issues in embedded systems code.

Care-taking Points:

- **Clarification of Self-Study Topics:** Doubts arising from self-study topics will be addressed and clarified during classroom sessions to ensure comprehensive understanding.
- **Hands-On Learning:** Practical exercises, real-world case studies, and projects will form an essential part of the course to help students apply theoretical knowledge to practical situations.
- **Industry-Relevant Tools and Techniques:** Students will work with industry-standard tools and technologies to develop the skills needed to solve real-world data science problems.

- **Continuous Evaluation:** Regular assessments, including quizzes, assignments, and project-based evaluations, will be systematically conducted to monitor student progress and reinforce conceptual understanding.
- **Expert/Guest Lectures:** Interaction with industry experts through guest lectures, webinars, or workshops will offer insights into the latest trends and applications in data science.

Charotar University of Science and Technology (CHARUSAT)

Faculty of Technology and Engineering (FTE)

Subject: Introduction to Computer Graphics (CEUE204) (Elective-I)

Semester: 4th

Teaching Scheme:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/Week	4	2	-	6	5
Marks	100	50	-	150	

Course Description:

The course introduces the fundamental concepts and applications of computer graphics, covering essential topics such as 2D and 3D transformations, drawing algorithms, surface modelling, and shading techniques. Students will explore the working principles of graphics hardware, software tools, and APIs, along with methods for representing and rendering geometric shapes and scenes. The course structure consists of lectures, tutorials, and laboratory work to provide practical insights into the field of computer graphics.

Course Objectives:

1. To gain a comprehensive understanding of the fundamental principles and applications of computer graphics.
2. To familiarize students with output primitives, basic drawing algorithms, and anti-aliasing techniques.
3. To learn and apply 2D and 3D geometric transformations, windowing, and clipping algorithms.
4. To explore surface modelling techniques using curves, splines, and parametric surfaces.
5. To understand the concepts of illumination, color models, shading techniques, and basic ray tracing.
6. To utilize graphics software tools and APIs for developing computer graphics applications.

Course Outcomes:

By the end of the course, students will be able to:

1. **CO1**: Understand the foundational concepts of computer graphics, including its applications and the hardware/software ecosystem.
2. **CO2**: Apply algorithms for drawing points, lines, circles, and basic shapes, including filled and unfilled primitives.
3. **CO3**: Perform and analyse 2D geometric transformations for graphical objects.
4. **CO4**: Apply transformations and viewing techniques for 3D graphical objects.
5. **CO5**: Understand and implement surface representation techniques for modelling.
6. **CO6**: Explain the fundamentals of light and shading to enhance realism in graphics.

Syllabus:

Unit No.	Unit/Topic Details		Hours (hr)	Evaluation Weightage (%)
1	**BASICS OF COMPUTER GRAPHICS**		08	14
	1.1	What is Computer Graphics (CG)?		
	1.2	Applications of CG: Gaming, simulations, CAD, medical imaging, and entertainment.		
	1.3	How CG works: Pixels, resolution, coordinate systems.		
	1.4	Graphics hardware: Display devices (CRT, LCD, OLED), input devices (mouse, keyboard, touch).		
	1.5	Graphics software: Overview of software tools and APIs (OpenGL, DirectX).		
2	**OUTPUT PRIMITIVES AND BASIC DRAWING ALGORITHMS**		12	22
	2.1	Pixel representation and connectivity (4-connectivity, 8-connectivity).		
	2.2	Line-drawing algorithms: DDA, Bresenham.		
	2.3	Circle-drawing algorithms: Midpoint and Bresenham's circle algorithm.		
	2.4	Polygon filling algorithms: Scan-line and boundary fill.		
	2.5	Anti-aliasing techniques for smooth edges.		
3	**2D GEOMETRIC TRANSFORMATIONS**		10	16
	3.1	Basic transformations: Translation, scaling, rotation, reflection, shear.		
	3.2	Composite transformations: Combining transformations using matrix representation.		
	3.3	Homogeneous coordinates for transformations.		
	3.4	Windowing and viewport transformations.		
	3.5	Clipping algorithms: Cohen-Sutherland line clipping, Liang-Barsky algorithm.		
4	**3D TRANSFORMATIONS AND VIEWING**		10	16
	4.1	3D geometric transformations: Translation, scaling, rotation about axes, reflection.		
	4.2	Viewing transformations: Perspective projections, Parallel projections (orthographic and oblique).		
	4.3	3D clipping basics.		
	4.4	Projection matrix and viewing pipeline.		
5	**SURFACE REPRESENTATION AND MODELLING**		10	16
	5.1	Curves and surfaces: Bezier curves, B-splines, cubic splines.		
	5.2	Parametric surfaces and surface of revolution.		
	5.3	Sweep surfaces and lofted surfaces.		
	5.4	Introduction to fractal curves and surfaces.		
6	**LIGHT, COLOUR, AND SHADING**		10	16
	6.1	Illumination models: Ambient, diffuse, and specular reflection.		

	6.2	Colour models: RGB, CMY, HSV.		
	6.3	Shading techniques: Flat shading, Gouraud shading, Phong shading.		
	6.4	Basic introduction to ray tracing.		
		Total Hours:	**60**	

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3	1	2	2	1	-	-	1	3	1	-	-	-
CO2	3	3	3	-	2	-	-	-	1	3	-	-	-	-
CO3	3	3	3	3	2	-	-	-	-	3	-	-	-	-
CO4	3	3	3	3	2	-	-	-	-	3	-	-	-	-
CO5	3	3	3	3	2	-	-	-	-	3	-	-	-	-
CO6	3	3	3	-	2	-	-	-	-	3	-	-	-	-

The correlation levels 1, 2 or 3 are as defined below:
1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High), If there is no correlation, put "-" placed.

Recommended Study Material:

- **Text books:**

1. "Computer Graphics with OpenGL", by Donald Hearn and Pauline Baker.
2. "Fundamentals of Computer Graphics", by Peter Shirley.

- **Reference Books:**

1. "Computer Graphics", by Donald Hearn and M. Pauline Baker.
2. "Computer Graphics: Principles & Practice in C", by J. D. Foley, S. K Feiner, A Van Dam F. H John, 2nd Edition.
3. "Interactive Computer Graphics: A Top-Down Approach with WebGL", by Edward Angel and Dave Shreiner.
4. "Mathematical Elements for Computer Graphics", by David F. Rogers and J. Alan Adams.
5. "Procedural elements for Computer Graphics", by David F Rogers.
6. "Computer Graphics", by Zhigand Xiang, Roy Plastock, Second edition.

Charotar University of Science and Technology (CHARUSAT)
Faculty of Technology and Engineering (FTE)

Subject: Fundamentals of Information Security (ITUE205) (Elective-I)

Semester: 4th

Teaching Scheme:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/Week	4	2	-	6	5
Marks	100	50	-	150	

Course Pre-requisites:

- Computer Networks and Internet Protocol: https://nptel.ac.in/courses/106105183

Course Description:

This course provides a foundation in information security concepts, including cryptographic techniques, network security, system security, and emerging threats. It emphasizes both theoretical understanding and hands-on implementations.

Course Objectives:

1. To develop a strong mathematical foundation for cryptographic techniques and security algorithms.
2. To explore symmetric and asymmetric cryptographic techniques for secure communication.
3. To understand message authentication, hash functions, and digital signatures for ensuring data integrity and authenticity.
4. To provide insights into system security, malware threats, firewalls, and internet privacy solutions like VPNs and Tor.
5. To encourage self-study and research-oriented learning through real-world security implementations and case studies.

Course Outcomes:

By the end of the course, students will be able to:

1. **CO1:** Apply the concepts of different techniques to implement security goals and security mechanisms.
2. **CO2:** Apply the different encryption and decryption algorithms using symmetric & asymmetric approaches to provide data confidentiality.
3. **CO3:** Apply the integrity and authentication aspects, like digital signature and message digest, and map them with practical use.
4. **CO4:** Apply web application, network, and system security to make them immune to attack.

- **Pre-requisites online video / course:**
  - Computer Networks and Internet Protocol, Prof. S. Ghosh and Prof. S. Chakraborty, IIT Kharagpur: https://nptel.ac.in/courses/106105183

- **Self-study/Further Study components and materials:**

Unit No.	Unit/Topic Details	
1	**MATHEMATICAL FOUNDATIONS FOR INFORMATION SECURITY**	
	1.1	Practical applications of Number Theory in cryptography
	1.2	Understanding Modulo Operations and its role in encryption
	1.3	Hands-on with OpenStego / OpenPuff for Steganography
	Materials: ➢ https://math.mit.edu/research/pure/number-theory.html#:~:text=Number%20theory%20has%20applications%20in,the%20arithmetic%20of%20K3%20surfaces ➢ https://embeddedsw.net/OpenPuff_Steganography_Home.html	
2	**SYMMETRIC KEY CIPHERS**	
	2.1	Comparative study: DES vs AES security
	2.2	Hands-on: Implementing AES encryption in Python
	2.3	Exploring Modes of AES Operations with practical examples
	Materials: ➢ https://www.khanacademy.org/computing/computer-science/cryptography ➢ https://www.youtube.com/watch?v=O4xNJsjtN6E	
3	**PUBLIC KEY CRYPTOGRAPHY**	
	3.1	Comparing RSA, ECC, and ElGamal cryptography
	3.2	Generating RSA Key Pairs using OpenSSL
	3.3	Understanding Public Key Infrastructure (PKI) and Certificates
	Materials: ➢ https://www.gpgfrontend.bktus.com/extra/algorithms-comparison/#:~:text=Use%20Cases%3A%20ElGamal%20is%20used,RSA%20or%20ECC%2Dbased%20methods ➢ https://www.ibm.com/think/topics/public-key-infrastructure#:~:text=Public%20key%20infrastructure%20(PKI)%20provides,%2C%20integrity%2C%20nonrepudiation%20and%20authenticity	
4	**MESSAGE AUTHENTICATION & HASH FUNCTIONS**	
	4.1	Hash Collisions: Understanding how MD5 vulnerabilities led to real-world attacks
	4.2	Hands-on: Generate SHA-256, SHA-512 hashes in Python
	4.3	Digital Signature verification using OpenSSL
	Materials: ➢ https://www.youtube.com/watch?v=b4b8ktEV4Bg ➢ https://aruljohn.com/blog/python-sha-md5-hash/	

Syllabus:

Unit No.	Unit/Topic Details		Hours (hr)	Evaluation Weightage (%)
1.	**BASIC CONCEPTS OF INFORMATION SECURITY**		05	8.33
	1.1	Introduction to Information Security: Terminologies, Goals of Information Security		
	1.2	Implementation Issues of the Goals of Information Security		
	1.3	Control Mechanisms for Information Security		
	1.4	Access Control - Administrative and Technical		
	1.5	Passwords - Are they secure?		
	1.6	Passwords, Hash Function, Common Password Threats		
	1.7	Multifactor Authentication – Challenges		
	1.8	Spheres of Information Security		
2.	**INTRODUCTION AND MATHEMATICAL FOUNDATIONS FOR INFORMATION SECURITY**		12	20
	2.1	Security Goals, Security trends - Attacks, Services and Mechanisms		
	2.2	Classical Encryption techniques (Mono-alphabetic, Poly-alphabetic substitution techniques and Transposition techniques) Overview of steganography along with demo tools like OpenPuff Tool, OpenStego, etc.		
	2.3	Number Theory - Prime And Relative Prime Numbers, Modular Arithmetic, Congruence, Fermat and Euler's theorem, Euclid's Algorithm, Chinese Remainder theorem		
3.	**SYMMETRIC KEY CIPHERS**		15	25
	3.1	Simplified Data Encryption Standard		
	3.2	Data Encryption Standard (DES), Triple DES		
	3.3	Block Cipher Principles		
	3.4	Characteristics of Advanced Symmetric Block Cipher		
	3.5	Block Cipher Design Principles		
	3.6	Advanced Encryption Standard Algorithm (AES)		
	3.7	Modes of Operations (Electronic Code Book, Cipher Block Chaining Mode, Cipher Feedback Mode, Output Feedback Mode & Counter Mode)		
4.	**PUBLIC KEY CRYPTOGRAPHY**		9	15
	4.1	Principles of Public-Key Cryptography		
	4.2	Public Key Cryptography Standards (PKCS)		
	4.3	RSA Algorithm		
	4.4	Key Management		
	4.5	ElGamal Algorithm		
	4.6	Diffie-Hellman Key Exchange		
	4.7	Elliptic Curve Cryptography (ECC)		
5.	**MESSAGE AUTHENTICATION AND HASH FUNCTION**		11	18.33
	5.1	Message Authentication		
	5.2	Hash Functions		
	5.3	Message Authentication Code (MAC)		

	5.4	Security of Hash Functions And MAC		
	5.5	Secure Hash Algorithm (SHA)		
	5.6	Hash-Based Message Authentication Code (HMAC)		
	5.7	Digital Signatures, Types of Digital Signature: RSA, Elgamal		
6.	**SYSTEM SECURITY, INTERNET PRIVACY WITH PROXIES, VPNS AND TOR**		8	13.34
	6.1	Malware, Functions of Malware, Sources of Malware, Layers of Defense Against Malware		
	6.2	Firewall Characteristics, Capabilities and Limitations, Types of Firewalls (Packet Filtering, Stateful Packet Inspection, Application Proxy & Circuit-level Proxy)		
	6.3	System administration and security, ufw UNIX firewall		
	6.4	Packet Capture demo using tools like Tcpdump, Wireshark, etc.		
	6.5	Wireless Network, Security Threats, Public Networks & Administering Wireless		
	6.6	Internet Privacy with Proxies, VPNs and Tor,		
	6.7	Recent Research Papers on Security		
		Total:	**60**	

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	1	1	-	1	-	1	1	-	-	-	1	2
CO2	3	3	3	2	3	1	-	-	-	-	-	-	1	1
CO3	2	2	3	2	3	2	-	-	2	1	-	-	1	1
CO4	3	3	3	2	3	-	2	1	2	1	1	2	3	3

The correlation levels 1, 2 or 3 are as defined below:
1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High), If there is no correlation, put "-" placed.

Recommended tools used:

1. Penetration Testing: Kali Linux, Parrot Security OS
2. Steganography: Steghide, StegoSuite, Xiao Steganography
3. Footprinting & OSINT: OSINT Framework
4. Port Scanning: Nmap
5. Cryptography & Encryption: OpenSSL, CrypTool
6. Vulnerability Assessment: OpenVAS
7. Password Recovery: Passware, Advanced Archive Password Recovery, Advanced PDF Password Recovery
8. Hashing & Data Integrity: HashCalc, MD5 Calculator
9. Firewall Security: Various firewall software solutions (to be evaluated)
10. Network Traffic Analysis: Wireshark, TCPdump

Recommended Study Material:

❖ **Textbooks:**

1. William Stallings, "Cryptography and Network Security Principles and Practice", 8th Edition – Pearson (2023).
2. Behrouz A. Forouzan, "Cryptography and Network Security", 3rd Edition - McGraw Hill Education (2015).

❖ **Reference Books:**

1. Atul Kahate, "Cryptography and Network Security", 4th Edition - The McGraw-Hill Companies (2019).
2. William Stallings, "Network Security Essentials: Applications And Standards", 6th Edition - Pearson Education (2018).
3. Douglas Robert Stinson, "Cryptography: Theory and Practice", 4th Edition - CRC Press (2018).

Online Resources:

1. Introduction to Information Security I, IIT Madras
   Link: https://nptel.ac.in/courses/106106129
2. Cryptography, Information Security
   Link:
   https://www.youtube.com/playlist?list=PLc4vStPmkiS6JPVys1KZ0UtSQDXu5ibNM
3. Ethical Hacking Essentials (EC-Council)
   Link: https://charusat.edu.in/ilms/course/view.php?id=157
4. Network Defense & Ethical Hacking
   Link: https://charusat.edu.in/ilms/course/view.php?id=159

Practical/Lab Work Implementation:

- **Penetration Testing OS Selection**: Evaluate and install the most suitable penetration testing OS (Kali Linux vs Parrot Security OS) based on user requirements, user-friendliness, and hardware requirements.
- **Steganography Implementation:** Use tools like Steghide, StegoSuite, and Xiao Steganography to conceal/hide secret files and messages in other file formats.
- **Footprinting & OSINT:** Gather target information using the OSINT Framework to analyze vulnerabilities and reconnaissance techniques.
- **Port Scanning with Nmap:** Perform scanning and enumeration using Nmap to detect open ports and services in a network.
- Implement public key encryption like RSA/ECC/Diffie Hellman for secure data transfer using OpenSSL.
- **Vulnerability Assessment with tools like OpenVAS:** Conduct vulnerability scanning of networks using OpenVAS and analyse security risks.
- **Password Recovery Techniques:** Recover encrypted passwords from applications using tools like Passware, Advanced Archive Password Recovery, and Advanced PDF Password Recovery.

- **Hash Verification for Data Integrity:** Generate and compare hash values using HashCalc and MD5 Calculator to ensure data integrity.
- **Cryptographic Analysis using tools like CrypTool:** Explore cryptographic algorithms and analyze their strengths using CrypTool.
- **Firewall Security Testing**: Evaluate and compare different firewall software solutions for securing network traffic.
- **Network Analysis:** Capture and analyze network traffic using tools like Wireshark or TCPdump etc.

Care Taking Points:

- **Clarification of Self-Study Topics:** Doubts arising from self-study topics will be addressed and clarified during classroom sessions to ensure comprehensive understanding.
- **Case Study-Based Pedagogy:** Case study-based learning will be the primary teaching method to provide practical exposure and hands-on problem-solving experience in security.
- **Hands-On Learning:** Practical exercises, real-world case studies, and projects will form an essential part of the course to help students apply theoretical knowledge to practical situations.
- **Industry-Relevant Tools and Techniques:** Students will work with industry-standard tools and technologies to develop the skills to solve real-world data security problems.
- **Regular Assessments:** To track progress, continuous assessment will be conducted through quizzes, assignments, case studies, presentations, and research or project work.
- **Expert/Guest Lectures:** Interaction with industry experts through guest lectures, webinars, or workshops will offer insights into the latest trends and applications in security.

Charotar University of Science and Technology (CHARUSAT)
Faculty of Technology and Engineering (FTE)

Subject: Nature And Properties of Materials (MEUE202) (University Elective – II)

Semester: 4th

Teaching Scheme:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/Week	-	4	-	4	2
Marks	-	50	-	50	

Course Description:

- This course introduces to the basics of metals and metallic alloys, polymers, composites and smart materials which have extensively broadened the scope of engineering design in the fields of Civil, Mechanical, Aerospace and other structural applications. After learning this course, students will be well- versed with the underlying principle governing the material properties and should be able to select proper material for their application.
- **URL**: https://nptel.ac.in/courses/112104203

***PRE-REQUISITES*:** Basic Physics and Mathematics Courses at the First Year Level, added with thirst for learning.

INDUSTRY SUPPORT: Every industry recommends to have a basic knowledge about various materials and truth behind their properties.

Syllabus:

Week 1: Introduction to Engineering materials & Mechanical properties

Week 2: Atomic bonding and crystal structure

Week 3: Metals and Ceramics

Week 4: Polymers

Week 5: Composite Materials

Week 6: Smart Materials

Week 7: Materials selection in Engineering design

Week 8: Non-mechanical properties and Laboratory demonstration

Recommended Study Material:

❖ **Text books / Reference Books:**

1. Material Science and Engineering: An Introduction by W.D. Callister, Wiley publication.
2. Engineering materials: Michael F. Ashby and David Jones, Elsevier publication.
3. Materials selection in Mechanical design by Michael F. Ashby, Elsevier publication.
4. Mechanics of Composite by Materials Robert M. Jones, Taylor & Francis publication.
5. Smart Structures Theory by Inderjit Chopra & Jayant Sirohi, Cambridge press.

Charotar University of Science and Technology (CHARUSAT)
Faculty of Technology and Engineering (FTE)

Subject: Solar Energy Engineering and Technology (EEUE202) (University Elective – II)

Semester: 4th

Teaching Scheme:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/Week	-	4	-	4	2
Marks	-	50	-	50	

Course Description:

- The course content is designed to provide comprehensive knowledge on solar radiation, analysis of solar radiation data, fundamentals of the solar thermal and photovoltaic system along with storage of energy required for effective design of efficient solar energy conversion devices. The concepts will be illustrated with practical examples, schematics and block diagrams wherever required. A sufficient number of numerical problems with solutions will be discussed in the course. This course is specifically designed for undergraduate and postgraduate students of Energy Engineering and Technology. Further, the course will be very much useful for students and researchers from varied academic backgrounds for the synthesis of novel energy conversion devices and processes.
- **URL**: https://onlinecourses.nptel.ac.in/noc24_ge51/preview

PRE-REQUISITES: Basic knowledge of heat transfer, thermodynamics and fundamentals of physics.

INDUSTRIES SUPPORT: This course will be very much effective for the engineres working in the solar industries.

Syllabus:

Week 1: Energy Scenario, overview of solar energy conversion devices and applications, physics of propagation of solar radiation from the sun to earth.

Week 2: Sun-Earth Geometry, Extra-Terrestrial and Terrestrial Radiation, Solar energy measuring instruments.

Week 3: Estimation of solar radiation under different climatic conditions, Estimation of total radiation.

Week 4: Fundamentals of solar PV cells, principles and performance analysis, modules, arrays, theoretical maximum power generation from PV cells.

Week 5: PV standalone system components, Standalone PV-system design.

Week 6: Components of grid-connected PV system, solar power plant design and performance analysis.

Week 7: Fundamentals of solar collectors, Snails law, Bougers law, Physical significance of Transmissivity - absorptivity product.

Week 8: Performance anlaysis of Liquid flat plate collectors and testing.

Week 9: Performance anlaysis of Solar Air heaters and testing.

Week 10: Solar thermal power generation (Solar concentrators).

Week 11: Thermal Energy Storage (sensible, latent and thermochemical) and solar pond

Week 12: Applications: Solar Refrigeration, Passive architecture, solar distillation, and ermeging technologies.

Recommended Study Material:

❖ **Text books / Reference Books:**

1. G. N. Tiwari, Solar Energy, Fundamentals, Design, Modeling and Applications, Narosa, 2002.
2. S. P. Sukhatme and J. K. Nayak, Solar Energy: Principles of Thermal Collection and Storage, Tata McGraw Hill, 2006.
3. C. S. Solanki, Solar Photovoltaics: Fundamentals, Technologies and Applications, Prentice Hall India, 2nd Edition, 2011.
4. J. A. Duffie and W. A. Beckman, Solar Engineering of Thermal Processes, John Wiley, 2006.
5. D. Y. Goswami, F. Kreith and J. F. Kreider, Principles of Solar Engineering, Taylor and Francis, 1999.
6. H. P. Garg and J. Prakash, Solar Energy: Fundamentals and Applications, Tata McGraw Hill, 1997.
7. M. A. Green, Third Generation Photovoltaics: Advanced Solar Energy Conversion, Springer, 2003.
8. A. Goetzberger and V. U. Hoffmann, Photovoltaic Solar Energy Generation, Springer--verlag, 2010.
9. K. Jager, O. Isabella, A. H. M. Smets, R.A.C.M.M. Van Swaaij, and M. Zeman, Solar Energy – fundamentals, technology and systems, Delft University of Technology, 2014
10. T. C. Kandpal and H.P. Garg, Financial Evaluation of Renewable Energy Technologies, McMillan India Ltd., 2013

Charotar University of Science and Technology (CHARUSAT)
Faculty of Technology and Engineering (FTE)

Subject: Ecology And Environment (CLUE202) (University Elective – II)

Semester: 4th

Teaching Scheme:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/Week	-	4	-	4	2
Marks	-	50	-	50	

Course Description:

- The objectives of the course is to introduce and sensitize all BTech students to the issue of ecology, environment and sustainability. The lectures are aimed at posing various questions that are relevant for all students of engineering and management to incorporate sustainability and a sensitivity to ecology and environment in their design of products, processes and systems.
- **URL:** https://archive.nptel.ac.in/courses/127/106/127106004/

Syllabus:

Week 1: Dr. B.S. Murty -Introduction (1), Sustainability Definition / Goals, Climate Change (2), Case Studies (3) (Eg: Dams, Chemicals, e-waste, IOT, Landfill siting etc)

Week 2: Dr. Sudhir Chella Rajan-Sustainability and Economics (3), Sustainability and Ethics (3)

Week 3: Dr. Ligy Philip-(Water Quality/ Waste Management), Water Quality and Treatment (3), Waste Management and Treatment (3)

Week 4: Dr. B. S. Murty (Water Management/ Resources), Urban Drainage, Water Resource Management, Impact of Climate Change

Week 5: Dr. Srinivas Jayanti (Energy)-Energy Demand / Resources (1), Pollution from Energy generation (1), Energy and Climate Change (Global Warming) (1), Energy and Sustainability (1), Long Range and Short-Range Solutions (1) (Global vs. India)

Week 6: Dr. R. Ravi Krishna-Risk Assessment Definition (1), Pollutant Pathways / Safety/ Exposure (1), Liability /Examples (1), Life Cycle Assessment (2), Environmental Management and LCA (1)

Week 7: Dr. Sudhir Chella Rajan-Urban Planning / Sprawl (1), Challenges in Urban Planning, Transport (1), Energy (Smart Grid) (1), Waste (1), Governance (1)

Week 8: Dr. Susy Varughese / Dr. Parag Ravindran-Ecology – definitions / Systems (1), Biodiversity (1), Examples of Historical Impact of economy on Ecology, Restoration / Ecological Engineering

Week 9: Dr. Ligy Philip / Dr. Ravi Krishna -Solid Waste Management, Hazardous Waste Management

Charotar University of Science and Technology (CHARUSAT)
Faculty of Technology and Engineering (FTE)

Subject: Introduction To Internet of Things (ECUE202) (University Elective – II)

Semester: 4th

Teaching Scheme:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/Week	-	4	-	4	2
Marks	-	50	-	50	

Course Description:

- Internet of Things (IoT) is presently a hot technology worldwide. Government, academia, and industry are involved in different aspects of research, implementation, and business with IoT. IoT cuts across different application domain verticals ranging from civilian to defence sectors. These domains include agriculture, space, healthcare, manufacturing, construction, water, and mining, which are presently transitioning their legacy infrastructure to support IoT. Today it is possible to envision pervasive connectivity, storage, and computation, which, in turn, gives rise to building different IoT solutions. IoT-based applications such as innovative shopping system, infrastructure management in both urban and rural areas, remote health monitoring and emergency notification systems, and transportation systems, are gradually relying on IoT based systems. Therefore, it is very important to learn the fundamentals of this emerging technology.
- **URL**: https://archive.nptel.ac.in/courses/106/105/106105166/#

PRE-REQUISITES: Basic programming knowledge.

Syllabus:

Week 1: Introduction to IoT: Part I, Part II, Sensing, Actuation, Basics of Networking: Part-I

Week 2: Basics of Networking: Part-II, Part III, Part IV, Communication Protocols: Part I, Part II

Week 3: Communication Protocols: Part III, Part IV, Part V, Sensor Networks: Part I, Part II

Week 4: Sensor Networks: Part III, Part IV, Part V, Part VI, Machine-to-Machine Communications

Week 5: Interoperability in IoT, Introduction to Arduino Programming: Part I, Part II, Integration of Sensors and Actuators with Arduino: Part I, Part II

Week 6: Introduction to Python programming, Introduction to Raspberry Pi, Implementation of IoT with Raspberry Pi

Week 7: Implementation of IoT with Raspberry Pi (contd), Introduction to SDN, SDN for IoT

Week 8: SDN for IoT (contd), Data Handling and Analytics, Cloud Computing

Week 9: Cloud Computing(contd), Sensor-Cloud

Week 10: Fog Computing, Smart Cities and Smart Homes

Week 11: Connected Vehicles, Smart Grid, Industrial IoT

Week 12: Industrial IoT (contd), Case Study: Agriculture, Healthcare, Activity Monitoring

Recommended Study Material:

- **Text books / Reference Books:**

1. S. Misra, A. Mukherjee, and A. Roy, 2020. *Introduction to IoT*. Cambridge University Press. Link: https://www.amazon.in/Introduction-IoT-Sudip-Misra/dp/1108959741/ref=sr_1_1?dchild=1&keywords=sudip+misra&qid=1627359928&sr=8-1
2. S. Misra, C. Roy, and A. Mukherjee, 2020. *Introduction to Industrial Internet of Things and Industry 4.0.* CRC Press.
   Link: https://www.amazon.in/dp/1032146753/ref=sr_1_3?dchild=1&keywords=sudip+misra&qid=1627359971&sr=8-3
3. Research Papers

Charotar University of Science and Technology (CHARUSAT)
Faculty of Technology and Engineering (FTE)

Subject: Software Conceptual Design (CEUE202) (University Elective – II)

Semester: 4th

Teaching Scheme:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/Week	-	4	-	4	2
Marks	-	50	-	50	

Course Description:

- This course will provide learners with an overview of what all is involved in creating conceptual software designs. Specifically, it will take learners through different aspects of understanding, creating and evaluating software conceptual designs.
- After going through the course, learners will be able to:
    1. Think of software in terms of sub-systems, and understand what issues have to be considered in order to design these sub-systems.
    2. Create a software conceptual design for a given design problem, and model them using Unified Modeling Language (UML) diagrams.
    3. Evaluate their designs for functional and non-functional quality attributes.
- **URL**: https://nptel.ac.in/courses/106101235

PRE-REQUISITE: Learners should have undergone a basic course in programming in any language, and should be familiar with basic programming constructs.

INDUSTRY SUPPORT: This course can be useful for graduates who wish to be employed in Software/IT companies in software engineering positions.

Syllabus:

Week 1: Deconstructing the software design process.

Week 2: Designing Software using the FBS Framework.

Week 3: Comprehending and Evaluating Software Designs.

Week 4: The Next Steps - Where does one go from here.

Recommended Study Material:

- **Text books / Reference Books:**

1. Software Engineering: A Precise Approach – Dr. Pankaj Jalote.
2. Cooperative Software Development – Dr. Amy Ko.

Charotar University of Science and Technology (CHARUSAT)
Faculty of Technology and Engineering (FTE)

Subject: Ethical Hacking (ITUE202) (University Elective – II)

Semester: 4th

Teaching Scheme:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/Week	-	4	-	4	2
Marks	-	50	-	50	

Course Description:

- Ethical hacking is a subject that has become very important in present-day context, and can help individuals and organizations to adopt safe practices and usage of their IT infrastructure. Starting from the basic topics like networking, network security and cryptography, the course will cover various attacks and vulnerabilities and ways to secure them. There will be hands-on demonstrations that will be helpful to the participants. The participants are encouraged to try and replicate the demonstration experiments that will be discussed as part of the course.
- **URL**: https://nptel.ac.in/courses/106105217

***PRE-REQUISITES*:** Basic concepts in programming and networking.

***INDUSTRY SUPPORT*:** TCS, Wipro, CTS, Google, Microsoft, Qualcomm.

Syllabus:

Week 1: Introduction to ethical hacking. Fundamentals of computer networking. TCP/IP protocol stack.

Week 2: IP addressing and routing. TCP and UDP. IP subnets.

Week 3: Routing protocols. IP version 6.

Week 4: Installation of attacker and victim system. Information gathering using advanced google search, archive.org, netcraft, whois, host, dig, dnsenum and NMAP tool.

Week 5: Vulnerability scanning using NMAP and Nessus. Creating a secure hacking environment. System Hacking: password cracking, privilege escalation, application execution. Malware and Virus. ARP spoofing and MAC attack.

Week 6: Introduction to cryptography, private-key encryption, public-key encryption.

Week 7: Cryptographic hash functions, digital signature and certificate, applications.

Week 8: Steganography, biometric authentication, network-based attacks, DNS and Email security.

Week 9: Packet sniffing using wireshark and burpsuite, password attack using burp suite. Social engineering attacks and Denial of service attacks.

Week 10: Elements of hardware security: side-channel attacks, physical inclinable functions, hardware trojans.

Week 11: Different types of attacks using Metasploit framework: password cracking, privilege escalation, remote code execution, etc. Attack on web servers: password attack, SQL injection, cross site scripting.

Week 12: Case studies: various attacks scenarios and their remedies.

Recommended Study Material:

- **Text books / Reference Books:**

1. Data and Computer Communications -- W. Stallings.
2. Data Communication and Networking -- B. A. Forouzan.
3. TCP/IP Protocol Suite -- B. A. Forouzan.
4. UNIX Network Programming -- W. R. Stallings.
5. Introduction to Computer Networks and Cybersecurity -- C-H. Wu and J. D. Irwin.
6. Cryptography and Network Security: Principles and Practice -- W. Stallings.

Charotar University of Science and Technology (CHARUSAT)
Faculty of Technology and Engineering (FTE)

Subject: Google Cloud Computing Foundations (CSUE202) (University Elective – II)

Semester: 4th

Teaching Scheme:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/Week	-	4	-	4	2
Marks	-	50	-	50	

Course Description:

- The Google Cloud Computing Foundations course aims to provide a detailed overview of concepts covering cloud basics, big data, and machine learning and where and how the Google Cloud Platform fits in. The course involves understanding concepts and perform hands-on training (via Qwiklabs platform) to practice the learning. There are 26 labs on Qwiklabs that are part of the course.
- Those enrolling for the course should ideally:
  - Have basic IT knowledge and be interested in learning more about Cloud and ML.
  - Have competency in at least one language (such as Python, Java).
  - Be familiar with the basics of shell scripting, SQL.
- **URL**: https://nptel.ac.in/courses/106105223

Syllabus:

Week 0: Introduction to the course

Week 1: So, What's the Cloud anyway? Start with a Solid Platform

Week 2: Use GCP to build your Apps

Week 3: Where do I store this stuff?

Week 4: There's an API for that! You can't secure the Cloud, right?

Week 5: It helps to network!

Week 6: It helps to network (continued)

Week 7: Let Google keep an eye on things. You have the data, but what are you doing with it?

Week 8: Let machines do the work

Recommended Study Material:

- **Text books / Reference Books:**

1. https://cloud.google.com/docs/
2. https://www.qwiklabs.com/

Charotar University of Science and Technology (CHARUSAT)
Faculty of Technology and Engineering (FTE)

Subject: Social Network Analysis (AIUE202) (University Elective – II)

Semester: 4th

Teaching Scheme:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/Week	-	4	-	4	2
Marks	-	50	-	50	

Course Description:

- Networks are a fundamental tool for modeling complex social, technological, and biological systems. Coupled with the emergence of online social networks and large-scale data availability in social sciences, this course focuses on the analysis of massive networks which provide many computational, algorithmic, and modeling challenges. The course will cover research on the structure and analysis of such large networks and on models and algorithms that abstract their basic properties. We will explore how to practically analyze large-scale network data and how to reason about it through models for network structure and evolution. Topics covered in this course are how information spreads through society; robustness and fragility of networks; algorithms for the World Wide Web; prediction and recommendation in online social networks; representation learning for large networks; etc.
- **URL**: https://nptel.ac.in/courses/106106239

PRE-REQUISITES: Python programming, Probability and Statistics, Machine Learning.

INDUSTRY SUPPORT: Any social media company, E-commerce company, etc.

Syllabus:

Week 1: Introduction; Tutorial 1: Introduction to Python/Colab; Tutorial 2: Introduction to NetworkX - Part I

Week 2: Network Measures; Tutorial 3: Introduction to NetworkX - Part II

Week 3: Network Growth Models

Week 4: Link Analysis

Week 5: Tutorial 4: Graph Visualization Tools; Community Detection - Part I

Week 6: Community Detection - Part II

Week 7: Link Prediction

Week 8: Cascade Behavior and Network Effects

Week 9: Anomaly Detection

Week 10: Introduction to Deep Learning; Graph Representation Learning - Part I

Week 11: Graph Representation Learning - Part II; Tutorial: Coding on Graph Representation Learning

Week 12: Applications and Case Studies; Conclusion

Recommended Study Material:

- **Text books / Reference Books:**

1. Social Network Analysis, Tanmoy Chakraborty, Wiley, 2021.
2. Network Science, Albert-Lazzlo Barabasi.
3. Social Network Analysis: Methods and Applications, Stanley Wasserman, Katherine Faus.

Charotar University of Science and Technology (CHARUSAT)
Faculty of Technology and Engineering (FTE)

Subject: Economics Of Health and Health Care (BMUD251) (University Elective - II)

Semester: 4th

Teaching Scheme:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/Week	-	4	-	4	2
Marks	-	50	-	50	

Course Description:

- When I was doing my doctoral research in the area of Health Economics several people asked me that What does this Economics in Healthcare mean?" Health economics or Economics of Healthcare is a discipline of economics which is concerned with the association between health status and the related resources assessing the value, behavior, efficiency and effectiveness of various stakeholders in the production and consumption of healthcare. Pertaining to several factors Health Economics has not been studied well in India or in many other developing countries, unlike the developed economies.
- **URL**: https://nptel.ac.in/courses/110104095

***INDUSTRY SUPPORT*:** Public Health Organizations, Hospitals, Allied Healthcare Providers, Pharmaceutical companies, NGO.

Syllabus:

Week 1: Introduction, Consumer Behaviour, Demand and Supply

Week 2: Elasticity of demand and supply, Theory of Production

Week 3: Theories of Cost and Grossman's Demand for Healthcare

Week 4: Market Imperfections, Healthcare Financing

Week 5: Health Insurance

Week 6: Impact Evaluation, Social Determinants of Health

Week 7: Economic Externalities, Public and Private Goods

Week 8: Population, Health and Development

Recommended Study Material:

❖ **Text books / Reference Books:**

1. Stephen Morris, Nancy Devlin, David Parkin. Economic Analysis in Health Care. John Wiley & Sons.
2. Folland, Goodmand, and Stano (FSG). The Economics of Health and Health Care. 5th Edition. Pearson Prentice Hall Press.
3. Jay Bhattacharya, Timothy Hyde, Peter Tu. Health Economics. Palgrave MacMillan.
4. Journals of Interest for Health Economists:
    - Journal of Health Economics; Health Economics; Health Economics Policy and Law; Journal of Public Health Policy; Health Policy and Planning; The European Journal of Health Economics; International Journal of Health Care Finance and Economics; Applied Health Economics; Health Policy; Social Science & Medicine; Pharmacoeconomics; Health Services Research; Journal of Health, Politics, Policy and Law; Health Affairs.

Charotar University of Science and Technology (CHARUSAT)
Faculty of Technology and Engineering (FTE)

Subject: Modern Application Development (CAUD204) (University Elective – II)

Semester: 4th

Teaching Scheme:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/Week	-	4	-	4	2
Marks	-	50	-	50	

Course Description:

- This course assumes that the student knows how to develop a typical monolithic desktop single user application. We gradually morph such a desktop app into a web app that uses distributed components and serves multiple users at the same time.
- In this process, we see how familiar ideas like user interfaces, data storage, and front-end vs backend distinctions get reformulated in the new web aware distributed context. We discover how the assumption of multiple users via multiple access points adds new concerns such as identity management and security, and learn techniques for addressing them.
- At the end of this course, you should be able to build a basic distributed web app and analyse constituent elements of the web and mobile applications that you use daily.
- **URL**: https://archive.nptel.ac.in/courses/106/106/106106222/

PRE-REQUISITES: Undergraduate Programming Experience

INDUSTRY SUPPORT: All Industries that use Computing

Syllabus:

Week 1 and 2: From desktop application to internet application

Week 3 and 4: Stateful applications

Week 5 and 6: The front end

Week 7 and 8: Databases and Simple files

Week 9: Setting up a website

Week 10: Using third party web services

Week 11 and 12: Extended project

Charotar University of Science and Technology (CHARUSAT)
Faculty of Technology and Engineering (FTE)

Subject: Ergonomics Workplace Analysis (PTUD192) (University Elective – II)

Semester: 4th

Teaching Scheme:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/Week	-	4	-	4	2
Marks	-	50	-	50	

Course Description:

- Ergonomic workplace analysis is a process where the ergonomic risk factors were evaluated using various validated tools and provide the probable recommendation to minimize the risk factors for development of work-related musculoskeletal disorders and improve the productive workday to reduce the cost for compensation, absenteeism and employee turnover. In the process of ergonomic workplace analysis, an ergonomist need to evaluate the physical work environment, psychosocial risk factors as well as various generic risk factors which leads to the development of work-related musculoskeletal disorders. This course is based on the complete process evaluation of EWA.
- **URL**: https://onlinecourses.nptel.ac.in/noc20_de12/preview

PREREQUISITES: Minimum Graduation with an understanding of Basic Ergonomics.

Syllabus:

Week 1: Overview of Ergonomics and understanding of Ergo-system Overview of ergonomic workplace analysis.

Week 2: Classification of techniques used in Ergonomics research.

Week 3: Subjective assessment tool and Methods Postural evaluation.

Week 4: Measurement of Work Effort and Fatigue.

Recommended Study Material:

- **Text books / Reference Books:**
  1. Introduction to Human Factors and Ergonomics, Robert S Bridger 4th Edition.
  2. Evaluation of Human Work, John R. Wilson, Sarah Sharples, 4th Edition, CRC Press.
  3. Handbook of Human Factors and Ergonomics Methods, Alan Hedge, Hal W. Hendrick, Karel Anton Brookhuis, Neville A. Stanton, CRC Press.
  4. A Guide to Human Factors and Ergonomics, Martin Helander, 2nd edition, CRC Press.
  5. Ergonomic Workplace Design for Health, Wellness, and Productivity, Alan Hedge, 1st edition, CRC Press.

Charotar University of Science and Technology (CHARUSAT)
Faculty of Technology and Engineering (FTE)

Subject: Mindfulness And Well-Being: Living with Balance and Ease (NRMD261)
(University Elective – II)

Semester: 4th

Teaching Scheme:

Teaching Scheme	Theory	Practical	Tutorial	Total	Credit
Hours/Week	-	4	-	4	2
Marks	-	50	-	50	

Course Description:

- This course offers a transformative exploration of mindfulness as a powerful tool for enhancing personal well-being, emotional resilience, and everyday balance. Rooted in evidence-based practices, participants will learn to cultivate present-moment awareness, manage stress mindfully, and foster compassion toward self and others. Through guided meditations, reflective exercises, and practical strategies, the course supports learners in integrating mindfulness into daily life promoting calm, clarity, and a deeper sense of ease. Ideal for individuals seeking to improve mental health, relationships, and overall quality of life in a fast-paced world.
- **URL**: https://www.coursera.org/learn/foundations-of-mindfulness-ii-living-with-balance-and-ease#modules

Syllabus:

Unit No.	Title of Unit	Prescribed Hours
1.	**Introduction to Mindfulness and well-being:** • Introduction	02
2.	**Stress and Mind/Body system:** • The Physiology and Health Consequences of Stress • The Stress Process • Habitual Reactivity to Stress • Awareness of Breath Guided Meditation • Body Scan • Stress and Wellness • Stress and the Brain • The Science of Stress and How Our Emotions Affect Our Susceptibility to Burnout and Disease • The Mind Body Interaction in Disease	05
3.	**Introducing Mindfulness:** • 3-Minute Breathing Space • Strategies for Cultivating Mindfulness	10

	• Mindful Stress Reduction • Awareness of Breath and Body Guided Meditation • Awareness of Breath Guided Meditation • Body Scan	
4.	**Creating and Sustaining Balance in an Unstable World:** • Fixed vs. Growth Mindset • Mindfulness and Equanimity • Awareness of Breath and Body Guided Meditation • Mountain Meditation	08
5.	**Achieving Positive Health:** • Optimizing Well-Being • Changing Behavior Mindfully • Compassion vs. Criticism • Loving-Kindness Meditation • R.A.I.N. Meditation	05
	Total:	**30**

Recommended Study Material:

- **Text books / Reference Books:**
  1. "The yoga Encyclopedia- Acomprehensive Dictniory of Yogic Concepts", by Dr Narain Prakash Giri, Dr Sunil Kumar, 2023.
  2. "Kriya Yoga- The science of life force", by Swami Nityanda Giri, 2018.
  3. "Mindfulness in Plain English", by Bhante Gunaratana, 2021.
  4. "Finding of Oasis Unveiling the intersection of Mind, Body and Spirit", by Sandeep Mall, 2024.
  5. "Mindfulness: A Practical Guide to Awakening", by Joseph Goldstein.
  6. "Vipassana: The Indian way to be Happy and Mindful", by Shonali Sabherwal.
  7. "The Headspace Guide to Meditation and Mindfulness", by Andy Puddicombe.